TOPICS IN TECHNOLOGY

SCOTT CARMICHAEL

Topics in Technology

Published by Scott Carmichael

Cover Art by Scott Carmichael

ISBN: 978-0-615-58087-6
(Part of Scott Carmichael's IT Manual: Topics in Technology, Linux Web Server Support, Information Technology Cautionary Tales and Help Desk Support Fundamentals)

Topics in Technology: An Introduction

I've wanted to write a book about technology for a long time, but never could figure out the best topic to cover. I initially toyed around with the idea of writing a programming book or perhaps a how-to guide for using popular graphic design applications. Ultimately though, I thought a "shotgun-approach" book would probably be the most useful and interesting type of title for the average reader – who perhaps wanted to know more about technology-related subjects they had heard about over the years, but never had a chance to further look into.

While I completely understand that the existence of sites like Wikipedia and countless online resources can always trump the content of any single book (in sheer volume of data available), I do believe that by personally going through dozens and dozens of sources while writing Topics in Technology (the vast majority of sources are other books, not mere web pages), readers will find this to be a very detailed, well-researched and comprehensive reading experience. The chapters were written to be short yet detailed, include proper citation information and cover a wide range of topics from all sorts of subjects relating to computer-based technology.

I hope you enjoy the book.

About the Author:

Scott Carmichael is an award-winning, professional Web Developer, Designer and IT Professional with degrees and certifications in both Visual Design & Web Media and Information Technology. *You can learn more about him and his work by visiting http://www.scottcc.info.*

Table of Contents:

Chapter 1: The History of Open-Source Software

It’s not uncommon nowadays for computer users to simply assume there are free software options available that offer many of the same features and capabilities paid-programs deliver. For example, if a user needs to do word processing, edit spreadsheets or even create presentations, they can either pay for the Microsoft Office suite of programs or they can choose to use the completely free Open Office set of applications from Oracle. Other examples would be choosing to use the free photo editor GIMP in place of Adobe’s Photoshop program or the Ubuntu Linux operating system instead of Windows 7 or Mac OS X.

Even though modern computer users have many choices in terms of free software solutions, this wasn’t always the case. In fact, before the 1980s, computer programmers encountered major problems with the sharing of code and collaborative development on projects due to the privatization, deregulation and protection of source code by profit-hungry, proprietary software-focused companies and individuals (Jones 52).

Among the programmers troubled by this situation was Richard Stallman, a Harvard educated programmer who also worked in the Artificial

Intelligence Lab at the Massachusetts Institute of Technology during the 1970s. Known as a “hacker,” Stallman left MIT over concerns regarding the university’s software copyright rules and personally believed in the free sharing and modifying of computer code (Britannica). Stallman publicly endorsed the idea of free sharing of code and software by giving his own applications away (namely the “legendary” Emacs editor he wrote in 1976) (Weber 49) to anyone who wanted a copy (provided they send him a tape to copy it on and a return envelope) (Jones 5-6).

In 1985 Stallman founded the Free Software Foundation, an organization originally created to focus on the development of a free alternative to Unix called “GNU” (GNU means “GNU’s not Unix”). Five years later he was awarded a MacArthur Fellowship and the no-strings-attached financial stipend he received allowed him to focus completely on his free software alternatives. Over the next few years he released the GNU Emacs Editor, GNU Compiler and GNU Debugger – software that would eventually become part of the Linux operating system (Britannica).

Stallman wanted to create a free operating system because he felt that “a modern computer required a proprietary operating system to do anything useful.” Unfortunately for end users, however, was the fact that the proprietary operating systems didn’t allow them to modify the OS source code. In Stallman’s mind, his proposed solution to this problem wasn’t so much that open source software would be “free” but rather it would be “freedom” for users wanting to have something besides closed-source software.

A simple description of “open source” code is that it’s public, open to modification and nonproprietary. Users have the freedom to run an open source program for any purpose, change it to suit their needs, learn how it works, improve and share the code and distribute it to others.

According to the semiofficial “Open Source Definition,” three features are essential to labeling software as open source. For starters, the source code must be distributed with the software and any costs associated with software must not exceed the cost of distribution. Secondly, anyone can modify it or create new software from it under the same terms as the original license. Lastly, anyone can redistribute the software for free without having to pay the author(s) licensing fees or royalties (Jones 5-6).

Another requirement about open source worth mentioning involves open source code’s relationship with proprietary code as specified by the General Public License (GPL). For example, open source code cannot later be turned into proprietary code. In fact, proprietary code cannot use open source code at all (and open source code can’t use proprietary code either). Also, it’s not permitted by the GPL for open source software to be included with a non-free program unless the combination of software is released as free software (Jones 48).

While open source software has seen growing adoption over the past few decades there are still problems that exist with it. Actually, even when the Free Software Foundation started, programmers had concerns over what was trying to be accomplished. Some found the idea of "free" software to have potential marketing issues while others found the label of "free" to be a negative thing altogether (despite Stallman's claim that "free" referred to "freedom," not price). In addition to that, programmers simply wanted to use what they knew worked and they had issues with the GPL's terms forbidding the combination of proprietary and open source code (Jones 52). Another issue that arose during the early days of open source development was that instead of devoting time and resources to the completion of the GNU operating system, GNU programmers spent so much time maintaining and porting existing code and adding new features, that the GNU OS kernel was never finished (Jones 49).

Because of the forming of the Free Software Foundation, creation of the General Public License and contributions by programmers like Richard Stallman, computer users now have the ability to choose and develop software that suits their needs without necessarily requiring the use of paid or free-yet-proprietary software and code.

Chapter 2: Ethics in the Digital Age

With the introduction of the personal computer, the mass adoption of the internet as a means for communication and the ever-increasing focus on technology in our personal and professional lives, computer users, companies and organizations have found themselves with brand-new “gray” areas to deal with in terms of ethics.

Because of all the technological advances and new technology-related things people have learned to live with, have the issues of security, safety and trust been properly addressed in this new, digital age? When it comes to the ethical handling of the vast amounts of electronic data available, are there any more gray areas to be mindful of? Can businesses and end users alike feel at ease when dealing with their private information in our computer-driven society … or is has the potential of technology proven to be something too complex and large-scale to keep in check?

For the city of San Francisco in July 2008, the Department of Technology and Information Services discovered how risky it was to let too much information be taken care of by one individual. The person with too much control was Terry Childs, a network administrator, who refused to hand over passwords to the city's FiberWAN network when confronted about his suspicious activity at work.

Law enforcement was ultimately brought into the situation and – after Childs was arrested for what constituted a denial of service attack – he handed over the passwords. Despite Childs ultimately giving the city the information they needed to run the network again (which at one point operated for 12 days without administrative control), the city ended up spending close to a million dollars to clean up the mess the network administrator caused during the crisis.

In April 2010 Terry Childs was found guilty of one felony count of denying computer services and – despite the lengthy incarceration time (he remained in prison from 2008 through his trial in 2010 on a $5 million dollar bond) – a clear motive was never given. Prosecutors, however, believe that network security was less important than job security to Childs and he wanted to make himself indispensable. (McMillan)

Sometimes, however, the questionable actions fall on the side of businesses … and in the late 1990s a very large and successful internet company called DoubleClick faced backlash from consumers and privacy groups over its actions.

The issue started when DoubleClick (an Internet advertising agency that specialized in targeted advertising on over 11,500 client sites) purchased Abacus Direct (a database marketer) in November 1999 for $1.7 billion dollars. DoubleClick initially planned to merge its own collection of user information with the newly purchased data to create something "beneficial for both its customers and the customers of the firms it served."

It didn't take long however for privacy groups to speak up – and by March 2000 DoubleClick announced plans to postpone linking the two databases. Privacy groups feared that because DoubleClick would essentially be able to track users by name and IP address – and then link that data (and other personal information) together – no good would come to the end user. This data would purely serve the interests of DoubleClick and its advertiser clients.

However, since the matching of browsing information and tracking of users by using ads and cookies is not illegal, DoubleClick revised their policy in August 2001 and gave itself a way of using the data they had whether privacy groups liked it or not. They instituted an 'opt-out' program for users (instead of an 'opt-in' one) which would require users to specifically deny DoubleClick access to certain data they collected. Once again, privacy groups argued the company was not doing enough to be mindful of the best interests of end users but this time their cries fell on deaf ears. (George 156-157)

Sometimes, questionable conduct relating to technology doesn't start off with deceptive actions at all. For Joanne Clark and Nancy Garrity, former employees of John Hancock Mutual Life Insurance, they felt they were wrongfully terminated based on what the company found in the two workers' business e-mail accounts.

Both Clark and Garrity had been with the company for a good length of time (two years and twelve years, respectively) and before their termination had engaged in sending sexually explicit e-mails to one another from joke sites, third parties and even Mrs. Garrity's husband. The trouble started for them when they forwarded this type of information to a coworker who then reported the email content to a supervisor. After they were fired, the two workers filed wrongful termination and invasion of privacy suits against the company.

The two fired workers argued that they believed their e-mails were private and the company had no right to read or go one step further and fire them based on what their employer found. To make matters worse for the two former workers, both plaintiffs knew they had to be careful about what they sent using company e-mails and their employer could view their messages through the company's intranet system at any time. Clark and Garrity ultimately did not win their case. (Jennings 278-279)

At the end of the day, the actions, roles and ideas of all parties working with technology can quickly create complicated, confusing and potentially problematic situations if everyone isn't careful. When the gray areas in these interactions are reached, end users, companies and organizations deal with each other all need to be very careful about how they proceed. That way, the most ethical outcome may be reached with minimal problems happening.

Chapter 3: Information Technology Outsourcing

The issue of outsourcing in the Information Technology (IT) industry has become a subject widely studied, discussed and documented over the last couple of decades. Given the recent global economic downturn, the issue is more relevant than ever with consumers and businesses trying to determine what the best combination of domestic workers versus more inexpensive foreign workers should be. Because of how technology-focused Americans' lives have become, this debated issue seems to have no definitive analysis or solution. Outsourcing technology-related work can greatly benefit the goods and services Americans receive and – at the same time – greatly affect (for good or bad) the quality and quantity of domestic workers in IT in general.

Part of the problem with keeping IT workers here in the United States is that success in a field like software engineering, IT support or networking requires a highly educated and talented workforce. Based on Americans' opinions about IT outsourcing, many seem to believe there is a shortage of engineers and scientists in the United States. (Macher and Mowery 85) Forrester Research, an American consulting company, predicts that 3.3 million

American service jobs will move offshore (most of these in IT) by 2015. (Carmel and Tjia 30)

Up through the late 1990s, there was a boom in the IT sector and this led to many Americans entering in Computer Science-related fields at the college level. After 2000, however, the number of students planning to enter technology-focused fields dropped dramatically. This may be explained by the Dot-Com Bubble bursting in the early 2000s or by decreased interest in general about Information Technology subjects by students. Regardless though, while Americans were losing interest in technology-focused careers over the past decade, there is evidence showing students entering Computer Science programs in other countries is rising. (Macher and Mowery 85)

From a business perspective, the attitude towards outsourcing in IT is being viewed more and more as a strategic necessity. If a company can use outsourcing to provide more options for revenue or lower their costs, competitors must match their behavior and output or fall behind. A good example of a company not participating in outsourcing and failing was Zenith Electronics, one of America’s largest television manufacturers. They avoided outsourcing for decades and eventually shrunk so much they completely disappeared. (Carmel and Tjia 10)

The main reason why companies use outsourcing is to decrease costs associated with software development or IT support. In a leaked IBM memo, the document revealed a typical Chinese programmer's cost to the business (including benefits and salary) is $12.59 an hour – about 20% of what an equivalent domestic programmer at IBM makes with 3-5 years experience.

Despite the large numbers of outsourced technology-related workers in India, the demand for cheaper IT help is so great that some American companies and even Indian firms are using Vietnam, China and other countries' workers because the wages there are even less. (Carmel and Tjia 33)

Even though outsourcing can help businesses, studies show that half of the largest corporations in America don't use it at all. When you factor in medium- and small-sized businesses and companies in Europe and Japan, the percentage is even higher. Many reasons exist for why companies avoid outsourcing but the most important reasons can be broken down into five areas.

First, when dealing with outsourced workers, the only means of communication is via telephone, e-mail or video conferencing. Some studies suggest 80% of a message is not in the text itself, but the delivery. Second, managers like to keep tabs on workers and ensure they are working well and effectively – if a worker is in another country, physically interacting with employees is impossible. At that point, management must rely solely on e-mail and telephones to express their concerns, solutions and ideas. Third, in software development – a field that requires a lot of minor and major adjustments – this type of work may be negatively affected by the absence of workers in close proximity to each other. Being nearby can often lead to better discussions about things that need to be addressed during projects. Fourth, workers who are physically near each other get to interact regularly in a more personal environment and this can lead to better communication skills and trust between employees. Finally, you must take into consideration all of the differences of domestic versus outsourced workers in the areas of values, beliefs, principles and behaviors associated with each side's culture. Any problem in these areas can lead to miscommunication, longer development time and may even cancel out any benefit there is to using outsourced workers. (Carmel and Tjia 12-13)

Unfortunately, even if potentially problematic issues with using outsourced workers in technology-related fields exist, the facts point to this trend among American businesses to continue. Lowering the cost of immigration or those associated with H1-B visas will not permanently bring talent from abroad stateside in the near future (in significant numbers) and the Government's stagnate interest in domestic Computer Science research funding in recent years is proving to be detrimental to IT-related academic programs (and current/potential students) in the United States.

In order for IT fields to be successful in the future (and remain that way), America's school and university systems need to play bigger, better and more important roles in teaching Computer Science, Mathematics and related subjects to United States citizens. (Carmel and Tjia 95)

Chapter 4: The History of YouTube

Even though it really only gained mainstream popularity after the mid-2000s, it's hard to imagine the internet now without people linking to, sending and watching millions of online videos all over the world on any given day. In what has become the largest and most well known video sharing web site, YouTube alone has changed the way the average American consumes video content online (Levy 2). But in the quest to deliver video content in new ways, YouTube and similar sites have found that turning this new media outlet into a steady source of revenue (and profit) isn't nearly as simple or straightforward as their creators initially planned.

YouTube's success story can be traced back to the early days of the internet. In 1987 when CompuServe's GIF image format was invented, average computers users found themselves with a relatively simple yet crude way of creating and viewing animations. It wasn't until around 1994 though when internet users started sending custom GIF-based animations to each other (and links to them) did the idea of "viral content" emerge. One of the most famous GIF-based "viral" animations from this early time in content sharing was The Hamster Dance (created in 1998) that featured, you guessed it – dancing hamsters. Viral video is described as a “video clip or recording, especially sent via e-mail and which gains widespread popularity through sharing.”

As technology progressed in both internet connection speed and hardware capabilities, so did the consumption rates for this new type of customized, shared content. Thanks to the Flash format, users started to share content that included sound as well as animation – a popular example of which is “Peanut Butter Jelly Time,” a short animated clip featuring a singing, dancing banana. When the Flash format gained the ability to have video content included in it, user-made home movies truly became the first wave of viral videos. Even before the days of YouTube, the video of the “Star Wars Kid” – a movie showing a Canadian teenager pretending to be a Star Wars character – was viewed just under 1 billion times. As you can tell from the examples listed above, viral content that could quickly spread across the internet didn’t need to be anything of great value or importance – it just had to be interesting to watch. YouTube co-founder Chad Hurley himself said this in regard to user-made content and professionally produced video (seen in TV shows and movies): “With easy and affordable access to cameras, editing software, and computing power, the playing field has been truly leveled.” (Levy 3-5)

YouTube, oddly enough, started out as an unsuccessful dating site and eventually turned into the video sharing network we see now. In 2006 Google purchased the site for $1.6 billion dollars (Levy 2) and had its #1 video sharing position cemented by Time magazine's "Person of the Year" article that same year. YouTube is so dominant now that it streams more videos every day than all of its major competitors (MySpace, AOL, etc.) combined. (Dean and Lastufka 2)

Unfortunately for YouTube, success in the video sharing market comes at a price. In 2009, analysts said that YouTube would probably lose $470 million that year – nearly a third of the price Google paid for it – due to enormous bandwidth, licensing and miscellaneous costs. (Cashmore) YouTube itself claims that it serves nearly a hundred videos per viewer to 135 million unique visitors each month. (Boutin) In short, for a site that largely grew because of its lack of advertising (with users expecting uninterrupted, unlimited video for free (Cashmore)), Google has learned that advertising is a necessity if it wants YouTube to ever be profitable.

Because of this issue, YouTube – even in its earliest days – began making efforts to work with content creators to draw traffic to the site but also allow YouTube to display advertising on partners' videos. In the deal, the content creator would get 55% of the ad revenue while YouTube would keep 45%. YouTube also discovered it was nearly impossible to screen all of the videos being uploaded to the site (nearly 10 hours of video are uploaded to YouTube every minute) and dealing with potentially copyrighted user-uploaded videos made mass-advertising on all content impossible to implement.

So, in exchange for working with YouTube, partners get a chunk of the ad revenue, branding options and additional design features – something regular users don't get. (Dean and Lastufka 196-197)

When it comes to larger entities – like television studios – the partnerships between them and YouTube is far less agreeable. While networks like NBC have a presence on YouTube (to share in some advertising revenue and protect against piracy), they don't want to lose complete control over their content and the ability to drive traffic to their own sites. So, in 2008, NBC and News Corp. debuted Hulu, a video sharing site that doesn't allow user-generated videos to be uploaded but gives viewers access to content (like full episodes of TV shows and feature-length movies) they can't find anywhere else. It also successfully introduced something YouTube couldn't implement – advertising – and has grown to be the third largest video sharing site in terms of views (approximately 300 million a month). (Burns 101-102)

In light of Hulu's success, YouTube has been desperately trying to find ways to monetize its seemingly never-ending supply user-generated content. This includes things like making users pay to rent movies (instead of being able to watch them for free (Boutin)) or putting unskippable, unmutable pre-roll ads before videos. This money-centric behavior is negatively being called the "Hulufication of YouTube" and to many, the reason why money can't be made off the site is obvious, (as a writer for the Mashable social media web site explains) because "no

one wants to advertise on your cat video.”
(Cashmore)

Chapter 5: Online Data Mining

Nowadays, online commercial, personal, educational and governmental web sites rely on relevant statistics more than ever for continued growth and prosperity. Due to the rapid expansion of the Internet over the past decade, it has become the largest publicly-available data source in the world. Because the Web is so large and always growing, users today can find information on virtually anything they want.

The dilemma for online content providers is the issue of whether it makes sense (financially or simply time-wise) for them to continue creating Web-based materials. The problem has less to do with the quality of the materials being presented than the sheer amount of "noise" on the Internet – which can come from one of two sources. One part of the noise issue comes from anyone being able to produce Web content. This leads to issues in quality control because a lot of sites feature misleading or low-quality information. The other part of this "noise" issue comes from elements on sites like advertisements, privacy policies, navigation links, etc. – all of which must be ignored for a user to find useful information. This is the type of situation where data mining is useful for online content creators.

Data mining is defined as “the process of discovering useful patterns or knowledge from data sources, e.g. databases, texts, images, the Web, etc.” For collected data to be useful, the discovered patterns must be understandable, possibly useful and valid. The main reason why data mining is done is so content providers can determine what future trends may be like for things such as site traffic, search engine relevancy, visitor response, advertising success rates, etc.

When starting the data mining process, individuals referred to as “data analysts” determine target data and appropriate data sources. (Liu 4-6) Most of time, these analysts have other responsibilities within companies and their time cannot be totally devoted to data collection and study. (Clifton 12) Once data has been collected, analysts must then “clean up” the information to remove abnormalities and “noise.” It’s also possible the information may contain unnecessary attributes or is simply too much for what is needed (or is easy to work with). Once the data has been filtered during the initial steps, analysts need to study the collected data to determine which discovered patterns are statistically useful. Analysts use different visual and evaluation methods for interpreting data in the final stage of data mining. Because of the sheer amount of information that is typically worked with in data mining, this iterative process takes many rounds to produce desired, adequate results. (Liu 6)

In order for data mining to be useful, the site that is being analyzed needs to be fairly stable in the sense that its content, presentation and related links (such as search engine ranking) need to be fairly comparable (before the data mining process) to what the content creator wants to add to the site in the future. If things change too much, predictions based on past visitor behavior have greater chances of not being very useful.

Some data mining solutions can take up to 18 months to produce enough accurate data for content creators to use it effectively (unfortunately the Web and user behavior can change significantly in just a few months). It should also be mentioned that with online data mining, the process is never done and gathering all the data available is impossible. There comes a time when one must weigh the data mining investment against the potential results generated by it.

An example of a successful online data mining project would first begin with dividing the different aspects (page views, referrers, site search, e-mail, metadata, video, etc.) into more reasonably sized chunks. Statistics would first be collected for general information like visitors, page views, time-on-site and referrers. Once monitoring tools have been put in place, analysts can begin collecting data from aspects like site search statistics – followed by e-mail, affiliate and paid search sources. Lastly, data analysts collect information about custom variables and metadata and then move on to multimedia content like

Flash, video, web apps and widgets. As each phase is completed, data analysts are able to make better decisions and generate more statistical data. The division of the site into more manageable chunks also allows for the greatest cost / benefit output relating to data mining. (Kaushik 306-307)

Data mining for multimedia content such as video is particularly difficult, simply because there aren't established benchmarks or standards in the analytics field. (Petrushin and Khan 461) Because of multimedia content's complexity compared to traditional web content, analysts must know the questions they want answered upfront since data mining preparation for it is more difficult coding-wise. (Kaushik 311) When data mining is done for multimedia content like video, analysts must consider user behavior (such as view counts) relating to it without taking into account the actual content being presented (because content quality is subjective). Another approach is to focus on one video and analyze all aspects of it (including content) to determine what makes it a success or failure in visitors' eyes. (Petrushin and Khan 461-462)

If data mining isn't working the way a content creator expects, it's often because they haven't put enough time, energy or resources into performing this process correctly. Data mining is something that needs to be done often and on a regular basis to produce the most relevant and valuable results and is useful in recommending (and implementing) future changes. (Clifton 12)

With the Web getting larger every day, Web-based data mining is becoming increasingly popular and important for all content providers. (Liu 2)

Chapter 6: Cloud Computing

As more and more companies change their business services and models to take advantage of the Internet, companies are demanding lower-priced and easier-to-manage alternatives to traditional in-house IT services, hosting and data offerings.

“Cloud Computing” is an increasingly popular alternative to having your own IT department and server(s) store your online data, web sites and web-based software online. The term “Cloud Computing” is actually not an easy thing to clearly define and many professionals working with technology seem to each have a different answer for what it is exactly. If that weren’t confusing enough, in 2008 the CEO of Oracle (Larry Ellison) said the term “Cloud Computing” was being applied to everything related to computers and simply overused.

The “idea” of Cloud Computing is a bit easier to describe though and is basically a metaphor for “the Internet.” When network diagrams are made, a cloud icon is typically the symbol for the Internet and meant to represent all of the miscellaneous online “stuff” that is meant to make a network function properly. It is also occasionally meant to take the place of services other people will handle so – instead of writing it all out – a cloud icon is sometimes shown to fill this role (this refers to outside companies handling aspects usually done in-house).

Without getting into the specifics that so many technical people can't agree on, Cloud Computing (in terms of functionality) usually refers to the ability for users anywhere in the world to access applications that are stored in data centers (which are often far away from them geographically). Another company other than yours hosts these applications and also manage software updates, pay for the servers required for hosting and then often passes the savings back to the companies using their services (depending on their contracts) – all of which makes Cloud Computing a very cost effective web hosting solution.

Keep in mind that if a company chooses to host their own sites internally, the company's budget must be adjusted on a regular basis to accommodate the consistent employment of IT staff, purchasing of hardware and software, increased telecom costs, security and backup concerns and the potential of downtime or disaster recovery solutions if something were to happen internally within the company's data center. Because of all these concerns, it should come as no surprise that more and more companies are shifting towards Cloud Computing-based hosting solutions for their web-based needs. (Velte, Velte and Robert 3-4)

Now, explaining all the "benefits" of Cloud Computing above does not mean traditional data center setups within businesses will or should go away. In certain situations, a physical data center within a company is the best way to reach a company's technology goals. (Hurwitz, Bloor and Kaufman 62) For starters, not everything belongs in a Cloud-environment. If you have applications that are extremely important to your company and/or those applications perform complicated tasks with other applications (that perhaps can't be put online), it wouldn't necessarily make sense to choose Cloud Computing in this setup. Also, if a small number of users will be using an application or service in a Cloud-environment, you may quickly find that the costs and potential complexity associated with having an outside company maintain those services for your business isn't really beneficial to your company's goals or your employees' performance. (Hurwitz, Bloor and Kaufman 273)

Also, one thing that is rarely discussed but can happen to even the largest Cloud-hosting providers is the potential for a third-party business hosting and running your web sites, databases, applications, etc. to go down. When that happens, not only will your company be unable to access or use these Cloud-based things … but if your company relies on internet traffic for revenue, even a small amount of down time can cost your business financially (and perhaps negatively affect your online reputation as a dependable site to deal with). For example, in 2008 Amazon's S3 cloud storage service stopped working and was down until technicians could determine what the problem was and fix it. This absence of service caused many sites to not work properly and some applications were unavailable for eight hours. To make matters worse, that was the second time in the same year that Amazon's S3 service failed. (Velte, Velte and Robert 5)

In the end, it comes down to what your company is planning to do with its data on the Internet and the goals of organization internally on a technical level. For instance, if security is a big issue, perhaps having an outside company holding your data is not ideal. Or, if you're working with a limited budget and can't afford full-time IT employees, Cloud-based hosting may be the best value for your business. Regardless of your opinion about Cloud Computing, there's no denying that it has become an increasingly popular option for companies concerned with costs and embracing technology to stay competitive. (Hurwitz, Bloor and Kaufman 62)

Chapter 7: Artificial Intelligence

In the field of computers and technology in general, the idea of Artificial Intelligence (A.I.) has been a concept scientists, philosophers, mathematicians and programmers have been trying to understand, implement and refine for hundreds of years.

Around 1623 the first known calculating machine was created by a German scientist named Wilhelm Schickard. In 1642 another more famous device known as the Pascaline was created by mathematician Blaise Pascal. This calculating device could both add and subtract values and Pascal said "the arithmetical machine produces effects which appear nearer to thought than all the actions of animals."

Another mathematician and philosopher named René Descartes who lived during the same era wasn't as convinced technology could truly mimic human behavior. His argument was that while the human mind may one day be recreated physically using devices similar to the calculating machine, those creations would be ruled entirely by physical laws and display no more "free will" than a rock "deciding" to fall towards the center of the earth.

While philosophers and mathematicians dating back to Ancient Greek times were responsible for early developments in A.I., Artificial Intelligence only became a legitimate scientific field with the mathematical formalization of areas like probability, logic and computation. (Russell and Norvig 5-7)

In 1925, a scientist named Arthur Samuel designed programs for the game of Checkers that would allow an automated player to perform at a strong amateur level of skill. His programming impressed the world in 1956 when a televised game demonstration showed his programming allowed the automated player to perform better than its creator. This breakthrough disproved the notion that computers can only do what they're told to do. (Russell and Norvig 18-19)

In terms of measuring human brain activity and trying to determine how it functions (in order to have a better understanding of Artificial Intelligence), there have been many advances over the past century relating to this issue. In 1929 the Electroencephalograph (EEG) machine was created and allowed for measurement of brain activity. Later on, devices like the functional magnetic resonance imaging (fMRI) machine began to help scientists (and doctors) map certain areas of the brain and brain activity to physical behavior.

The human brain and computers may have similar characteristics that allow for comparison in certain aspects but the two are still very different. For instance, while the brain has a cycle time a million times slower than a computer, the brain has more connections and storage capacity than a typical high-end personal computer. Supercomputers, however, may have a capacity similar in scope to a brain's but capacity isn't everything – the brain does not use all of its neurons simultaneously which implies that "intelligence" is not equivalent to sheer capacity. In fact, even with a virtually unlimited amount of capacity, researchers still don't know how to reach the human brain's level of intelligence. (Russell and Norvig 11-12)

When people think of A.I. nowadays it's not uncommon to visualize humanoid robots and talking computers that mimic human behavior. For example, even though toy robots have been around since the 1950s, robotic toys with serious computing power only arrived in the 1990s. Virtual pets – small, pocket-sized electronic devices – combined the functionality of a digital clock with a digital game using "pets" that needed to be fed, taken to the bathroom, put to sleep, etc. (Jefferis 14) During this time an Australian scientist named Rodney Brooks created a robot named Cog that featured arms, a torso and a head. His goal was to mimic human interaction with the environment, in hopes Cog would "learn" about its surroundings the way humans do. Cog "sees" with cameras for eyes and "hears" with microphones. (Jefferis 12)

In 1999, Sony brought robotic animals to the average consumer with their robodog devices. Each successive version was more sophisticated than the last and could display behavior such as wagging its tail or barking a greeting. In addition to the built-in actions, programmers could also “teach” these robodogs new tricks by introducing additional code into their systems.

A more modern example of robots aimed at average consumers is the Roboraptor – a toy robot that resembles a dinosaur. This particular toy features touch sensors on its chin and mouth to allow the dinosaur to gently bite, along with microphones to detect where sound is coming from. At one time, only the highest-end research labs would have had access to a device containing functionality and equipment now found in the inexpensive Roboraptor toy. (Jefferis 14)

When it comes to Artificial Intelligence, because of the human brain’s complexity it’s hard to accurately gauge how “far advanced” researchers and scientists have come to understand it. The human brain has about 100 billion brain cells and work in ways that are just starting to be understood. Problems such as how the brain can quickly sort such large amounts of data still haven’t been solved and are the roadblocks to truly understanding how to create genuine A.I. in man-made technology. (Jefferis 10)

Chapter 8: Nanotechnology

Science fiction books, shows and movies often bring up the topic of nanotechnology (technology built at extremely small sizes) at some point in their stories to make it clear that what is being told is beyond the reach of current technology, science and medicine. But why is this so? What makes nanotechnology so elusive to modern-day scientists and engineers?

Nanoscience is the study of structures and molecules that are between 1 and 100 nanometers in size (1 nanometer is 1 billionth of a meter). This is seen by scientists and engineers as an "edge" of sorts between the normal world of physics and scientific laws we can all see, understand and physically interact with and the molecular and atomic world where quantum effects and wave-particle duality start changing the way things occur.

When thinking of objects (such as man-made devices) at a nanometer scale, it's important to remember that it is the point at which the largest molecules of living organisms and the smallest human-made technology can be created. As a point of reference, a human hair measures 50,000 nanometers across and the tiniest features commonly etched on computer chips in the early 2000s were around 130 nanometers in size. Nanoscale engineers aim to control the transfer of biological, chemical and physical materials between atoms and molecules in order to produce desired results – which are essential to creating functional “devices” at this level.

It should be noted that nanotechnology is often confused with its “sister” field, microelectrico-mechanical systems (MEMS) however the two are very different. In place of simply trying to work with the raw materials found at nanoscale sizes, MEMS engineers and scientists focus their attention on applications like tiny robots with mechanic arms that could be used to deliver drugs, repair damaged tissue or move through the body’s bloodstream. Along with medical uses, these tiny mechanical devices could be used to create or maintain larger mechanical devices which could then have completely different attributes or uses from the original technology that formed it. While the MEMS description may sound more like the stuff of science fiction, the point of nanoscience is still practical in both areas – it is the scientific study of and interaction with things at the nanometer scale (although MEMS are generally designed with the idea of operating at 1,000 to 1,000,000 nanometers). An example of a currently-working, real-world application of MEMS is the one used to trigger airbags in cars during accidents. (Ratner and Ratner 6-8)

In terms of what nanotechnology can do for people, a good example is the use of it in medical treatment. For instance, when a medical implant is trying to duplicate the characteristics of normal human bone, the implant must be penetrated by the tissue around it to strengthen it. Because the metal alloys used in implants wear out fast and the current materials used can’t be penetrated by tissue, expensive and frequent surgery is often needed. A solution to this problem would be using nanocrystalline zirconia ceramic – a durable, tough, bio-compatible material that also resists corrosion. Using this along with another material known as nanotitania allows the man-made implant to create a chemical bond with living bone (although the bond is fairly weak). Not only would these nano-made materials feature more benefits than existing materials they would also create implants that would need less replacing and last longer. (Wilson, Kannangara and Smith 81-82)

While the use of nanotechnology and MEMS on a large scale in everyday life is still the stuff of science fiction, nano scientists and engineers are finding new ways to understand and develop man-made creations at nanometer sizes on a daily basis.

Chapter 9: Social Technology

Even just a few years ago, if you mentioned the term "social technology" or "social networking" to someone, it's very likely you would have been met with a blank stare. Over the years though, average people using the internet have discovered social networking sites such as MySpace, Facebook and Classmates.com that allow them to communicate with thousands of people at the same time. These sites allow users to express themselves and connect with other people of similar interests. (Goetchius 5-6) Also, even though the internet can be described as very fragmented interest-wise, people in general still want to be aware of what the majority of people online are up to. (Qualman 32)

One of the main reasons why social networking has become so popular is that traditional personal web pages are often too complicated, expensive and/or time intensive for the average person to create and maintain. MySpace, Facebook, Friendster and other social networking sites give users opportunities to create profile pages which include built-in functionality like access to e-mail, forums, blogs, instant messaging and other communication tools. The profile pages on social networks can be used for personal or professional purposes and the connections between them are what create the social networks.

(Goetchius 5-6) It should be mentioned though that while social networking often includes the aspect of e-mail communication, kids today feel like e-mail is undesirable if they can respond with one-to-many updates via Facebook and MySpace. (Qualman 47)

Social technology is not only limited to traditional websites, however. Virtual worlds like Linden Lab's Second Life use 3D visuals combined with text/voice chatting and instant messaging features to give users a completely different online social experience. Users in Second Life create their own avatars – 3D representations of themselves that can take the form of virtually whatever they want (such as a super model or even a dinosaur) – and are only limited by their imaginations and technical know-how. In fact, everything in Second Life is created by the users (called residents) from waterslides to sunglasses to shopping malls.

One of the benefits to using a virtual world like Second Life is its ability to display graphics in a way that provide the users with completely different experiences from simply reading text on websites. For example, instead of meeting up with friends to attend a concert in real-life, a quick invitation (called a teleport request) will instantly transport the virtual world resident to the event where they can participate with the live show in real-time. In addition to attending concerts, residents can take online classes, go shopping (using the in-world currency known as Linden) and even party with other users at dance clubs.

As mentioned above, because of the in-world money being circulated (not only can the currency be used to buy in-world goods and services – it can also be converted into actual U.S. and foreign currency) and popularity of Second Life, real-world companies have taken a strong interest in the virtual environment. Residents can find virtual shops for Nike and American Apparel as well as corporate offices for businesses like Reuters and IBM. (Bell and Robbins 7-9)

Along with providing average computer users the opportunity to be part of online social communities, companies have also embraced social technology as a means for generating revenue and increasing brand awareness. Real-world print newspapers have encountered major financial problems because of the popularity of blogs and the countless qualified authors who provide relevant, accurate and compelling content for free. While this hurts the newspaper industry, it helps online sites carve out niche environments where smaller companies or writers can thrive online. In the case of newspapers, trying to quickly and easily convert an old business model to one made for online audiences simply does not work. (Qualman 32)

For other companies like Proctor & Gamble, social networking sites provide a way for them to get their products in front a targeted, engaged audience in new ways. When Proctor & Gamble launched a sister product of Secret deodorant called Secret Sparkle they used the MySpace profile pages of popular teen actresses to run ads promoting contests about the new product. The information gathered through the entries was then turned into a valuable mailing list for the company that could be used for future products they wanted to promote. Also, film studios have learned the value of creating online profiles for fictional characters (such as Jason Bourne or Ron Burgundy) as a form of advertising. (Goetchius 7-8)

Even with the plethora of possibilities in regards to creating and maintaining social connections using the internet, social technology in general is not without its flaws. For example, studies have shown that the writing ability of teens has been negatively affected by influences from social networking services. When surveyed by the non-profit group that administers the SATs, 50% of teens admit to using improper punctuation and capitalization in assignments, 25% have used emoticons in class work, 38% have used instant messaging jargon such as “LOL” and 64% have performed at least one of these offenses in an academic setting. (Qualman 54) It’s also been reported that students have been kicked out of school, people fired from work and teachers asked to step down for inappropriate actions and content displayed on social networking sites. (Qualman 33)

From their first appearance in 1995 to what we have available now, social networking sites and technology have given the average person the ability to connect with others in ways not possible even a couple of decades ago. (Goetchius 6)

Chapter 10: Video Games

Video games have become an increasingly popular form of entertainment over the last several decades with more and more people devoting time, energy and money to the hobby. For instance, in the early 2000s (when Sony's PlayStation 2 game system was being released in the United States), it was estimated that Sony's PlayStation 1 hardware was found in 27 million homes domestically (one in every six households) and 75 million worldwide. (Berger 4)

People play games mainly because it is a form of entertainment that temporarily lets the individual take a break from real-life and deal with fictional situations that have no or little consequences in the real-world. Not only do games allow a player to escape into an environment that is separate from the real-world, these games often have fictional/fantasy settings with their own rules that would be impossible to experience in real-life. (Berger 12)

One interesting thing to keep in mind when thinking about video games is that no one needs to play them. Unlike a software program a worker at a company needs to use in order to remain employed, games are programs that are made solely to attract – and keep – someone's attention. Because of the competitive nature of the game industry in which the vast number of games released can't be realistically played by all consumers, developers must aim to create games that are similar enough to what gamers are used to (to ensure financial success) and offer enough new enhancements to satisfy the creativity demands of consumers. (Bryant and Vorderer 9)

In efforts to make games appeal to audiences, game developers have learned that creating games within different genres helps categorize the industry's offerings in a more easily understood way. For example, puzzle games, first-person shooters, simulation games, sports games and role-playing games are just some of the many different types of genres that exist. Games within the same genre often feature the same or similar controls, graphics and game play designs – all of which help consumers know what to expect before buying and playing them. (Bryant and Vorderer 2)

When it comes to making the games more appealing to players, developers – as mentioned earlier – need to constantly update the presentation of titles in regards to graphics, sound and controls. In Medal of Honor: Frontline (a shooter), the player experiences force-feedback via a rumble feature in the controller to accompany the World War II game play, sounds and on-screen visuals. In modern games, developers now have the ability to easily include realistic voices in their games also – something that was hard to do in older, earlier video games in which programmers had to resort to crudely sampled or synthesized solutions. The addition of characters within games having the ability to speak allows players to experience auditory arousal – something researchers have to found to trigger affective responses when combined with realistic visuals and accurate controls.

In fact, the most important aspect of "being absorbed by a video game" is making the game seem "real" to a player. This is most effectively done with the use of highly detailed, realistic graphics. Like every form of entertainment, content creators aim to transport viewers (or in this case, gamers) to an alternate, controlled reality and have them believe in what they are seeing and doing there. The fewer "non-real" artifacts there are within these worlds, the more believable and authentic the fictional environments become. In the case of video games, realistic or highly-polished/stylized visuals provide gamers with more visual data to process and this, in turn, creates greater, more focused engagement from the players. (Bryant and Vorderer 45)

John Carmack, owner of id Software and the lead programmer responsible for Doom (a famous shooter), (Time Magazine) said the game was a "really significant inflection point for things, because all of a sudden the world was vivid enough that normal people could look at a game and understand what computer gamers were excited about." (Bryant and Vorderer 38) So not only must games appeal to people who already classify themselves as gamers, developers must aim to create new titles all the time that make non-gamers interested in their offerings.

For non-gamers, many find it easy to be overwhelmed by the industry and everything in it. Even researchers have had a hard time analyzing the field in general or games by themselves. Studies that focused on showing the patterns produced by people playing games can often have very different results. For instance, while some studies have players familiarize themselves with a game before attempting to play it, other studies give expert players no chance to learn about a game's design or mechanics before recording their observations. While neither way is incorrect in terms of studying player behavior, both can produce very different findings.

Also, when trying to measure the relationship between game difficulty and player skill, research has shown that player skill can have a great impact on the gaming experience. In terms of racing games, when novices start playing, they often gravitate towards vehicles with average stats across a wide spectrum of characteristics in order to get a "feel" for the game. As they become better at the game (through observation and repetition), players give up average capabilities to take risks with more demanding vehicles that require greater precision but offer better rewards. In other words, game content can be perceived very differently based on the skill of the player. (Bryant and Vorderer 69)

Video games have become a very important form of revenue for businesses and entertainment for millions of people all over the world. It's an industry where creativity, attention to detail, user feedback, player engagement, high production values and innovation is almost always rewarded (for both the players buying games and the business selling them) in perhaps the most logical way possible, compared to any other form of entertainment that has ever existed.

Chapter 11: Mainframe Computers

When people hear the term "mainframe computers," they probably envision a ridiculously expensive, room-sized machine sitting in a sterilized environment and doing the work of a thousand workers with little or no human input. While this description may have elements of truth to it (for instance IBM's z9 109 mainframe computer is over six feet tall and weighs over 2,600 lbs.), (Parsons and Oja 18) mainframe computers came into existence for a reason and aren't even thought of the same way anymore (among technology professionals and businesses) like they were upon their introduction decades ago.

From the 1950s to the 1970s, mainframe computers dominated the computational powered aspects of the technology industry. These giant machines often took up space on multiple floors of military and civilian research buildings and were relatively few in number around the world (compared to the number of personal computers today). (Goldberg 280)

A mainframe computer is an expensive and generally large computer that processes information for hundreds of thousands of users or tasks simultaneously. They are often operated in environments where security, centralized control and reliability are of the utmost importance. Governments as well as businesses use these devices to process, manage and store data on a large-scale. (Parsons and Oja 18)

In the 1960s, mainframe computers were largely proprietary in nature – in that they often were created using hardware and software that were incompatible with offerings from other manufacturers. (Dilligan 20) Not only were the mainframe systems expensive to purchase (a modern day mainframe computer normally costs at least several hundred thousands of dollars but can easily go above a million) (Parsons and Oja 18), but businesses typically had to pay for an in-house team of programmers to maintain and use them. Because programming staff job security was based on the performance and usefulness of the mainframes they were responsible for, they developed an obvious conflict in interest regarding the design and cost of mainframe technology. For example, as mentioned earlier, back in that time the hardware and software mainframe computers used varied greatly between manufacturers and that led to many interoperability issues between systems. Part of this problem was due to the differences between programmers in charge of using mainframe computers (because CPU and memory resources were so expensive, programmers had to find different methods for getting machines to do exactly what they wanted and programmers' solutions could be radically different) but most of the problem was due to the manufacturers' need to grow their market share in an intense business environment.

Unfortunately for businesses back then, once a manufacturer had a company as their client, that business was often stuck with them because everything relating to mainframe computers (hardware, software, physical resources and staff) was extremely expensive (due to everything – including software and perhaps even data – being proprietary). In fact, IBM had invested so heavily in mainframe technology, software and services that it purposely gave up portions of the market in order to avoid anti-trust issues. This led to companies like Digital Equipment Corporation and Cray Research getting a foothold in niche mainframe computing market segments relating to engineering, academic and scientific work. (Dilligan 20-21)

In the 1970s personal computers were introduced and quickly grew to outnumber mainframe computers. Not only were the personal computers far cheaper than mainframe computers, they could perform a greater variety and number of tasks when thought about collectively. Because of the rising importance of compatibility among personal computers, standardization started to occur in all areas of computer-related technology.

By the time the 1980s came around mainframe computers and personal computers were working together in ways that weren't previously possible. It became commonplace to think of computer-powered tasks as being solved in a distributed setup instead of relying on a single, powerful mainframe machine. This evolution in mindset and industry eventually resulted in a negative change of attitude towards mainframe computers and lowered their popularity among users in businesses, organizations and governments. In modern technology times, mainframe computers are thought of as “dinosaurs” compared to the popularity and potential of networked personal computers. (Goldberg 280)

While the need for largely singular, super-powerful computers will probably never go away (especially in areas like research, large-scale businesses or at government levels) it does seem as though the industry has changed so radically that the concept of “many personal computers” has become more favorable in most situations than a single “mainframe” machine.

Chapter 12: The History of Oracle

Oracle has grown into one of the largest and most profitable technology companies in the world thanks to its presence in the database market and its many acquisitions over the years. As of 2009, the company was valued at approximately $93 billion and employed over 86,000 employees. (Clark and Worthen)

The history and success of Oracle can be traced back to the 1950s and 1960s when computers first started using bulk storage methods and devices. The “classic medium” of storing and accessing data back then was using 2,400-foot long reels of magnetic tape and paper- or punched-cards. Back then, if a user wanted to access specific data on a reel, they would have to access it serially – that is, completely read the tape from start to finish or until the target data was reached.

Unfortunately, this way of accessing data forced programmers to develop complex records and methods of saving and updating information. For example, if an employee’s personnel information was stored on tape and a change needed to be made, you would need a tape (and tape drive) for the previous master file, a tape (and tape drive) for the new master file to be written and a third tape (and tape drive) that stored changes that were to be made.

Because information was stored serially and managing data was difficult, there may have been little more than an employee ID number or date to connect data records to one another. And if records-within-records existed, the same information may occur repeatedly (with slight variations or none at all) for every record stored on tape. And due to the design and implementation of serial, tape-based storage methods, the devices were a common cause of bugs in applications and often caused record corruption which could go weeks or months before being detected. (Ensor, Nørgaard and Gorman 5-6)

While working at IBM Research Laboratory in 1970, Dr. Edgar F. Codd, a graduate of Chemistry and Mathematics, began looking at different possible ways of making computer data queries. His work pointed him in the direction of Relational Calculus, a 19th century mathematical method of working with data. While his paper, "Relational Model of Data for Large Shared Data Banks" was important, it was largely ignored at the time until the early 1980s due to databases being "generally not quite complex enough" to justify the use of a relational database structure compared to, say, using traditional tape-based solutions. (Ensor, Nørgaard and Gorman 8-9)

In the 1970s Larry Ellison and the other co-founders of what would become Oracle Corporation started a company so that they could bid on government technology jobs. They were all familiar with the work of Dr. Codd and a new language called SQL (Structured Query Language) that aimed to make data interaction in databases faster and accurate than ever before.

Because the CIA had a need for computer-based solutions relating to data storage and manipulation, they invested in Ellison's company, SDL (Software Development Laboratories), and the company started working on the top-secret project, code-named Oracle. While the government project was ultimately canceled, Ellison and the co-founders quickly realized the commercial potential of relational databases and adjusted the company's business strategy. The company was renamed Relational Software, Inc. and in 1978 they developed the first version of Oracle.

In 1979 RSI released the first commercially available relational database product that used SQL and successfully beat IBM to the market for relational database software. (Abbey, Corey and Abramson 5-6) Recent reports from within Oracle say that there was never a "Version 1" and that the first official release was named "Version 2" so clients would view it as a more mature product than it actually was. (Ensor, Nørgaard and Gorman 10) At the time, IBM had an 80% marketshare in the database business. One reason for Oracle's early success was that it ran on a minicomputer which was significantly cheaper than mainframes and it quickly adapted to the C programming language (versus staying in Assembly, which meant code was different on every machine).

In 1980 the company was renamed again to Oracle System Corporation (and even later to Oracle Corp.) and due to its adoption of the C language (which allowed the source code to be used on multiple platforms and hardware), by 1983 hit $2.5 million in sales. In 1984 sales increased to $13 million and a year later reached $23 million. In 1985 Oracle decided to enter the financial application business in an effort to increase their possible commercial opportunities. (Abbey, Corey and Abramson 6-7)

At the time, IBM released their first relational database product called SQL/DS in 1983 (although some reports say 1981) and second relational database product, DB2 that same year. Due to IBM's slow-entry into the market, Oracle was able to get a sizable lead in the relational database field. Larry Ellison often joked in the 1980s about how Oracle software was, "plug compatible six years before IBM built the plug" (Ensor, Nørgaard and Gorman 10)

By 2000 Oracle had reached annual sales of over $10 billion and by that point had switched gears from being competitive with IBM to being a direct competitor with Microsoft. Its adoption of the Linux operating system and Java programming language cemented their desire to ensure there was as little overlap as possible with Oracle and Microsoft regarding offerings and revenue. In 2000, at the height of internet boom phenomenon, Larry Ellison passed Bill Gates as being the richest man in the world. (Abbey, Corey and Abramson 11-12)

In 2009 Oracle acquired Sun Microsystems Inc. for $7.38 billion in an effort to get a foothold in the hardware/server and programming language marketplaces. After the announcement, Larry Ellison was quoted saying Java was "the single most important software asset we have ever acquired." (Clark and Worthen)

While the company continues to be generating billions of dollars in revenue each year, in 2010 Oracle Corp. was sued by the U.S. Justice Department and accused of defrauding the government on a $1 billion software contract. (Kendall and Worthen) While this action will most likely not affect the immediate future of Oracle in any significant way, it is ironic to see the most successful database technology company ever get sued by the U.S. Government, the same entity that helped establish Oracle (and even provided its name) over three decades ago.

Chapter 13: The Java Programming Language

The Java programming language was created in rather humble beginnings in the early 1990s by Sun Microsystems employees trying to develop the "next big thing" in consumer devices. The goal of these employees (labeled internally as the "Green Team") was to create a new type of device that would allow users to interact with it in unique ways. After 18 months of research and development they produced the "Star7," a demonstration device much larger than a modern PDA that featured voice-input and touch screen controls. The team had envisioned the new platform being used in everything from kiosks to smart appliances to televisions. Unfortunately, while their idea was innovative, Sun had a hard time convincing industries (like digital cable companies) to implement it – in short, the Star7 was simply ahead of its time.

Luckily for Sun though, not all was lost. James Gosling, a member of the "Green Project" team, was the one responsible for the creation of the new language developed specifically for the Star7 device. Named "Oak," the language allowed the code to function on any type of processor and worked well within networked environments. When the Internet's popularity exploded in the mid-1990s, Sun's language, now labeled “Java” (due to a programming language named “Oak” already existing) (Greanier 2-3), gained a lot of popularity due to its bundling with Netscape’s browser, (Jackson) its ability to handle dynamic content and operate across networks. While the language can be found in all sorts of modern-day devices (smart cards, PDAs, cellphones, etc.), it has also made its way into large scale e-commerce and enterprise applications.

Concerning the technical details of Java, it can be described as both a platform and a language. Java’s secure, object-oriented, portable, high-performance and multi-threaded design allowed it to reach world-wide usage and popularity over the years. Additionally, it is often thought of as easier to learn, yet just as capable as languages like C and C++. One of the developers’ main goals was to remove the overly complicated and unnecessary parts of C and C++ from the new language to make it easier to work with. (Greanier 1-4) In fact, because Java’s syntax is so similar to that of C and C++, many programmers have a fairly easy time understanding it. (Horstmann and Cornell 3) Even JavaScript programmers find the transition to Java a rather simple one, due to the syntax being so similar. (Greanier 4) And when compared against C++, Java made many programmers happy by greatly reducing the potential for bugs within its code. This was accomplished by not having a programmer deal with things like memory allocation, pointers and garbage collection issues.

If that wasn’t reason enough to use Sun’s language, Java’s most obvious advantage was that regardless of your operating system – Windows, Macintosh, Linux, Solaris, etc. – your Java code would work on all of them without needing to be rewritten for each one. Also, another big reason why Java can be found in so many environments and devices is that it only needs a very small footprint to operate. The basic interpreter and class support is approximately 40KB in size and adding thread support and basic standard libraries pushes Java up to just 215KB. (Horstmann and Cornell 3-5)

Sadly though, not all things related to Java are positive. In a report from O’Reilly in 2008 that used programming book sales as a measurement for interest in languages, the data indicated that there had been a massive 50% drop in Java language interest over the previous five years. Meanwhile, in that same period, languages such as Ruby, C# and JavaScript gained consumer interest. In fact, O’Reilly predicted that within the same year, C# interest and sales would overtake Java if gains and declines in both languages’ book sales continued to occur (respectively). While this data alone is not necessarily proof positive that Java is going away, it’s clear that dynamic, web-based languages are becoming increasingly popular. (Asay)

Another interesting development in the Java story was the departure of its creator, James Gosling, from Oracle in early 2010. In January of the same year Oracle finalized its purchase of Sun Microsystems and with it gained official control over the Java platform and language. While James was quoted as saying “As to why I left, it's difficult to answer: just about anything I could say that would be accurate and honest would do more harm than good” and that he worried about Java’s recent, increasing community issues, he made it clear that “Oracle has certainly been incredibly committed to keeping Java and the whole ecosystem as strong and as healthy as can be.” The long-term effects of the acquisition are still to be seen, but with Gosling, Tim Bray (co-inventor of XML) and Jonathan Schwartz (Sun CEO) leaving so soon after Sun Microsystems’s purchase by Oracle, it certainly raises some questions about Java’s future. (Jackson)

However, Java is clearly not going anywhere anytime soon. According to a Dice.com statement from 2010 (Dice is an IT job board website), the most popular programming skill in demand is Java/J2EE and it is represented in 14,000 positions across the country. On top of that, Dice’s Senior Vice President Tom Silver said that these types of positions typically pay $10,000 more than the average technology professional’s annual pay. He later stressed that, “there are just not enough developers on the market and too many openings,” – words that should make Java programmers everywhere happy to hear. (Sears)

Chapter 14: E-Commerce

By the time the Internet started to gain widespread popularity in the 1990s, it became obvious that the potential of online commercial opportunities were going to be a focal point for Internet users and traditional off-line customers as well. (Botha, Bothma and Geldenhuys 9) Because of increased internet access in the 1990s, e-commerce (which is most commonly described as a site that allows customers to buy services and products (Reynolds 13)) saw dramatic growth during the later part of the decade. (Vulkan 11) By the early 2000s, over a billion mobile and wireless devices were in use, further expanding the reach of businesses to customers, regardless of location. (Botha, Bothma and Geldenhuys 67)

It’s worth pointing out, however, that early on many companies simply focused more on the publicity and brand awareness of being “first out of the gate” companies (like Amazon.com happened to be), rather than actual profitability. (Vulkan 11)

But just because you can make a website for selling goods and services doesn’t necessarily mean you should. A poorly implemented online business model can lead to financial problems, along with damage to a company’s brand and even their customer base. For example, during 1999’s holiday season, both Toys ‘R

Us and KBKids websites encountered problems that led to disastrous revenues and customer experiences. According to a survey performed by Robertson Stephens (a well known Equity Investment, Sales/Trading and Research company), 44% of surveyed customers in early 2000 said that their bad shopping experiences with KBKids.com would keep them from shopping on the site ever again. On the flip-side, Bloomingdale's website – which was a much smaller site at the time – fared much better during the same holiday season.

Also, companies need to make sure their e-commerce presence makes sense in many different ways. For instance, years ago 3Com embraced the Internet in a very logical and responsible way. Its e-commerce plan focused on getting its products and services to a global audience, providing an online store, making educational/technical support materials available and storing things such as drivers online (so customers could have instant and easy access to them). 3Com managed to use the Internet in a way that did not negatively affect its traditional ways of doing business, but rather enhanced them overall. (Reynolds 10) And because online businesses can operate 24 hours a day, customers can search for company offerings at any time and the entire shopping process can be automated in a cost effective way.

Please keep in mind, however, that "e-commerce" existed before the web gained popularity. Even in the early 1980s, businesses like Hewlett-Packard and General Electric were using computers and networks to trade digitally using EDIs (Electronic Data Interchanges). The problem, however, was that these companies often had their own hardware and software and suppliers had to share costs and use whatever software companies like HP and GE allowed them to. Thanks to the nature of the Internet and its use of standardization of web browsers (like Internet Explorer), compatibility between companies' and customers' transactions are no longer nearly as expensive or compartmentalized. (Vulkan 12-13)

Another important aspect of e-commerce to discuss is the process of transferring money between customers and businesses. Early on, customers were very leery about personal financial information being transmitted electronically but this issue has lessened due to the introduction of financial frameworks designed specifically for e-commerce payment transactions. And with the growth of mobile technologies support worldwide, consumers are more interested than ever in the idea of shopping online, anywhere.

When a customer actually pays for goods or services online, a lot of different things come into play. The first is that e-commerce sites must accept some form of payment, typically credit, charge and debit cards. (Botha, Bothma and Geldenhuys 67) The second is that transactions must be secure, and this most often done using SET (Secure Electronic Transaction) and SSL (Secure Socket Layer) forms of data encryption. (Reynolds 13) When it comes to actually processing e-commerce credit card payments, third party services (that charge monthly fees) are widely used. This is due partly to the potential of fraud (relating to payment promises and falsified order information) and that the storing of personal customer information is risky because of online anonymous parties and potential illegal activities involving them. (Botha, Bothma and Geldenhuys 68, 72) The third step is that details of the order must be stored electronically, usually in a database or in tab-delimited text files so that they can be properly invoiced. The final part of the process involves the use of a fulfillment steps that ultimately provide a customer with the goods and/or services they ordered. (Reynolds 13)

E-commerce is just a newer platform for selling goods and services that allows companies to have a wider reach and be more efficient in different ways. However, in order for organizations to be successful, they will still require the use of traditional ways of dealing with customers and doing business. (Botha, Bothma and Geldenhuys 9)

Chapter 15:
The History of Silicon Graphics

Silicon Graphics, Inc. (SGI) – a company best known for its high-end 3D-software focused hardware – was a business that took the graphics marketplace by storm in the 1990s, only to collapse under its own weight and poor decisions in the late 2000s. SGI was founded by James Clark (a professor at Stanford University) in 1982 when he saw the need for better graphics processing on computers. His creation, the "Geometry Engine," allowed hardware to deal with complex code that could generate realistic 3D imagery.

The company became a success due to its business model that effectively took the spotlight away from supercomputers that once had to do this type of work (this was also the way Sun Microsystems, Inc. managed to become a major technology company). Hardware and software that may have once cost a company $250,000 could be purchased from SGI for $75,000. While Silicon Graphics originally undercut competition to gain market share, eventually they became so used to charging a premium for their products that they priced themselves out of the ever-changing industry. (Vance 67)

Some of the major problems started for Silicon Graphics in 1994 when they unveiled their plan to purchase Wavefront Technologies and Alias Research, two software companies that developed "entertainment graphics software." Because of their dominance in the field of high-end graphics technology, monopolistic concerns were raised and SGI was required by the FTC to do certain things with the software, to keep the playing field level for all companies. The requirements to port the software to non-SGI platforms (from Sun, HP, IBM, etc.) and create APIs (Application Program Interfaces) for compatibility affected SGI's new purchases' potential because the software could not be exclusive to SGI hardware. (Hope 119)

It also didn't help that the rise in personal computer popularity (and lowering prices of them) came about in the 1990s. While the company enjoyed selling workstations in the early 1990s for $15,000 or more (which were very popular in the auto industry, Hollywood, etc.) by late in the decade their dependence on an older revenue system (generated from high priced hardware sales) had done them too much harm overall.

In an attempt to stay relevant, SGI acquired super-computer competitor Cray Research in 1996 for $740 million. (Poletti) Despite the elimination of a rival, Silicon Graphics made a poor decision in selling off the server business of the new purchase to Sun Microsystems for just tens of millions of dollars … and it was that server business that eventually went on to make billions annually for its new owner.

Another blunder SGI made was focusing on the Linux Operating System and Intel's new Itanium processor when the company – in an effort to stay competitive – decided to stop using its own proprietary software (IRIX) and hardware (MIPS). While Linux faced competition from Microsoft's Windows, Intel's Itanium didn't deliver as expected, causing Silicon Graphics even more problems.

Finally, the reduced spending by companies in the early 2000s and the Dot-Com crash certainly added more to SGI's worries, simply because Silicon Graphics had focused so much and for so long on selling costly hardware and software … and couldn't seem to adapt to changing times.

Following a bankruptcy in 2006 in which the company almost went under (in 2005 SGI had reduced its workforce to less than 2,000 employees) (Vance 67-68), on April 1st, 2009 the company announced it was going into bankruptcy yet again and selling itself for a mere $25 million (the buyer was Rackable Systems, Inc. – a company that makes data servers). At the time of its second bankruptcy, SGI had $527 million in debt and 1,100 employees (which was still considered to be "bloated" for what the company made and revenue generated). (Poletti) And when you rewind back even further, you'll realize SGI's finances had been in trouble for a long time. While the company generated $3.66 billion in revenue in 1997, by 2004 it was down to just $824 million. Also, when you factor into the equation that SGI sold Cray Research for $100 million just four years after purchasing it for over $700 million, the writing was clearly on the wall that Silicon Graphics was no longer growing by the late 1990s. (Seba 52)

The future of the company is still uncertain, mostly because Rackable Systems hasn't made it clear which parts of SGI it thinks are valuable. The drive to purchase SGI was made by Rackable Systems in an effort to gain new customers and get "in the low-end segment of the high performance computing arena," according to Thomas Weisel Partners. Up to this point, Rackable had faced challenges from companies like Dell in the server market and had been struggling on their own as well (their revenue went from $351 million in 2007 to $247 million in 2008).

What was once a hugely successful company best known for creating the technology used to make visuals for films like Jurassic Park and Terminator 2 is now just a mere shadow of its former self, thanks to a string a bad business decisions, increased competition and the lowering prices of technology. (Poletti)

Topics in Technology: Sources

Abbey, Michael, Michael J. Corey and Ian Abramson. Oracle9i: A Beginner's Guide. McGraw-Hill Companies, Inc., 2002.

Asay, Matt. C# set to take Java's crown as Java drops 50 percent. 3 March 2008. 8 August 2010 <http://news.cnet.com/8301-13505_3-9884500-16.html>.

Bell, Mark and Sarah Robbins. Second Life for Dummies. Wiley Publishing, Inc., 2008.

Berger, Arthur Asa. Video Games: A Popular Cultural Phenomenon. Transaction Publishers, 2002.

Botha, J., P. Bothma and P. Geldenhuys. Managing E-Commerce. Juta & Co., 2004.

Boutin, Paul. YouTube expands video rentals. But audiences aren’t flocking. 23 June 2010 <http://venturebeat.com/2010/05/04/youtube-rental/>.

Britannica, Encyclopædia. Richard Matthew Stallman. 6 June 2010. 6 June 2010 <http://www.britannica.com/EBchecked/topic/1055658/Richard-Matthew-Stallman>.

Bryant, Jennings and Peter Vorderer. Playing Video Games: Motives, Responses, and Consequences. Lawrence Erlbaum Associates, Inc., 2006.

Burns, Kelli S. Celeb 2.0: How Social Media Foster Our Fascination with Popular Culture. Greenwood Publishing Group, 2009.

Carmel, Erran and Paul Tjia. Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce. Cambridge University Press, 2005.

Cashmore, Pete. YouTube Ads: You Hate Em. 23 June 2010 <http://mashable.com/2009/04/05/youtube-ads-youhate-em/>.

Clark, Don and Ben Worthen. Oracle Snatches Sun, Foiling IBM. 29 April 2009. <http://online.wsj.com/article/NA_WSJ_PUB:SB124022726514434703.html>.

Clifton, Brian. Advanced Web Metrics with Google Analytics. Wiley Publishing, 2010.

Dean, Michael W. and Lastufka, Alan. YouTube: An Insider's Guide to Climbing the Charts. O'Reilly Media, Inc., 2009.

Dilligan, Robert J. Computing in the Web Age: A Web-Interactive Introduction. Plenum Press, 1998.

Ensor, Dave, et al. Oracle Insights: Tales of the Oak Table. Apress, 2004.

George, Richard T. De. The Ethics of Information Technology and Business. Blackwell Publishing Ltd., 2003.

Goetchius, Alex. Career Building Through Social Networking. Rosen Publishing Group, Inc., 2008.

Goldberg, Elkhonon. The New Executive Brain: Frontal Lobes in a Complex World. Oxford University Press, 2009.

Greanier, Todd. Java Foundations. SYBEX Inc., 2004.

Hope, Einar. Competition Policy Analysis. Routledge, 2002.

Hurwitz, Judith, et al. Cloud Computing For Dummies. Wiley Publishing, 2010.

Jackson, Joab. Java founder James Gosling leaves Oracle. 10 April 2010. 8 August 2010 <http://www.computerworld.com/s/article/9175218/Java_founder_James_Gosling_leaves_Oracle?source=rss_news>.

Jefferis, David. Robot Brains. Crabtree Publishing Company, 2006.

Jennings, Marianne M. Business: It's Legal, Ethical, and Global Environment, Seventh Edition. Thomson/South-Western, 2006.

Jones, M. Tim. GNU/Linux Application Programming. Charles River Media Inc., 2005.

Kaushik, Avinash. Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity. Wiley Publishing, 2010.

Kendall, Brent and Ben Worthen. Justice Department Sues Oracle for Fraud. 30 July 2010. <http://online.wsj.com/article/SB10001424052748703578104575397573983263294.html>.

Levy, Frederick. 15 Minutes of Fame: Becoming a Star in the YouTube Revolution. Penguin Group, 2008.

Liu, Bing. Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data. Springer Science+Business Media, 2007.

Macher, Jeffrey T. and David C. Mowery, National Research Council (U.S.). Committee on the Competitiveness and Workforce Needs of U.S. Industry, National Research Council (U.S.). Innovation in Global Industries - U.S. Firms Competing in a New World. The National Academic Press, 2008.

McMillan, Robert. Admin who kept SF network passwords found guilty. 27 April 2010. 8 June 2010 <http://www. networkworld.com/news/2010/042710-admin-who-kept-sf-network.html>.

Parsons, June Jamrich and Dan Oja. New Perspectives on Computer Concepts 2010. Course Technology, 2010.

Petrushin, Valery A. and Latifur Khan. Multimedia Data Mining and Knowledge Discovery. Springer Science+Business Media, 2007.

Poletti, Therese. Goodbye SGI: We knew you too long. 2 April 2009. 14 December 2011 <http://www.marketwatch.com/story/sgis-rise-fall-a-silicon-valley>.

Qualman, Erik. Socialnomics: How Social Media Transforms the Way We Live and Do Business. John Wiley & Sons, Inc., 2009.

Ratner, Daniel and Mark Ratner. Nanotechnology: A Gentle Introduction to the Next Big Idea. Prentice Hall, 2003.

Reynolds, Janice. The Complete E-Commerce Book. CMP Books, 2004.

Russell, Stuart J. and Peter Norvig. Artificial Intelligence: A Modern Approach, Third Edition. Prentice Hall, 2010.

S., Horstmann Cay and Cornell Gary. Core Java 2: Fundamentals. Sun Microsystems Press, 2001.

Sears, Don E. Programming, Development Skills in Demand. 26 July 2010. 8 August 2010 <http://www.eweek.com/c/a/IT-Management/Programming-Development-Skills-In-Demand-505665/>.

Seba, Tony. Building a Winner – 9 Rules of Business and Marketing Success from Silicon Valley's Wealth Machine. Tony Seba, 2006.

Time Magazine. TIME Digital -- Digital 50 - John Carmack. 1999. 1 August 2010 <http://www.time.com/time/digital/digital50/10.html>.

Vance, Ashlee. Geek Silicon Valley: The Inside Guide to Palo Alto, Stanford, Menlo Park. Morris Book Publishing, LLC, 2007.

Velte, Toby, Anthony Velte and Elsenpeter Robert. Cloud Computing: A Practical Approach. McGraw-Hill, 2010.

Vulkan, Nir. The Economics of E-Commerce. Princeton University Press, 2003.

Weber, Steven. The Success of Open Source. Harvard University Press, 2004.

Wilson, Mick, et al. Nanotechnology: Basic Science and Emerging Technologies. University of New South Wales Press Ltd., 2002.

TOPICS
IN
TECHNOLOGY