



An e-newsletter published by
Software Quality Consulting, Inc.

Fall 2012
Vol. 9 No. 2

Welcome to **Food for Thought™**, an e-newsletter from **Software Quality Consulting**. I've created free subscriptions for my valued business contacts. If you find this newsletter informative, I encourage you to continue reading. To receive this newsletter via e-mail, visit www.swqual.com and enter your e-mail address for a free subscription..

Your continued feedback on this newsletter is most welcome. Please send your comments to steve@swqual.com

This Month's Topic

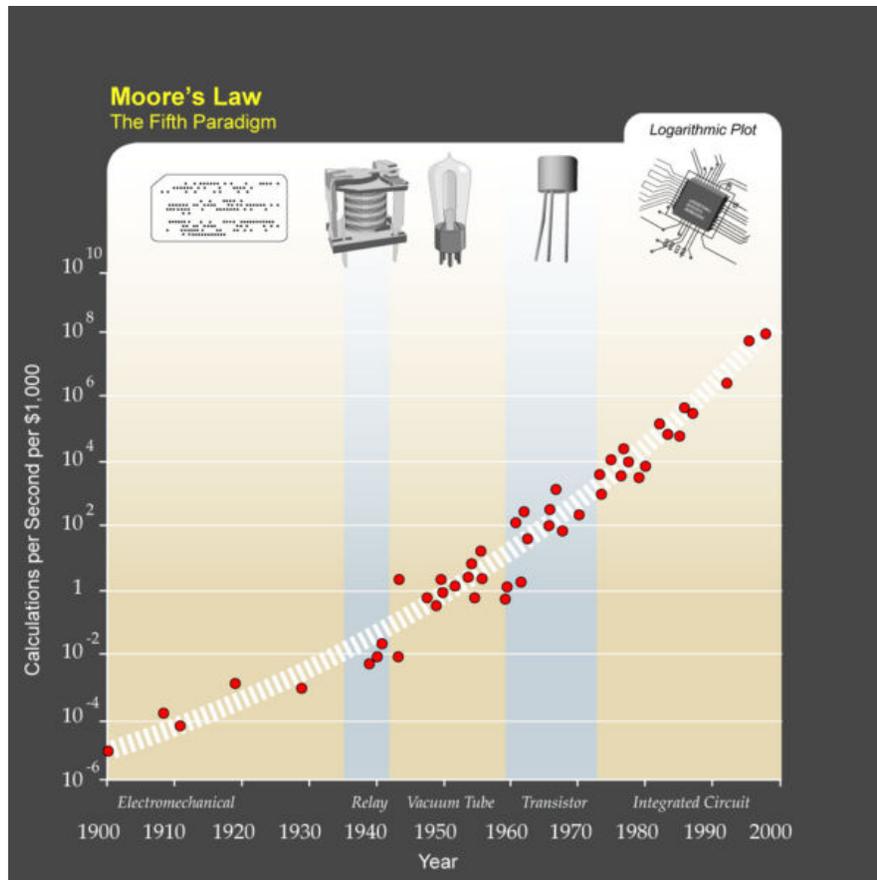
Have you become technically irrelevant?

It's been four decades since I graduated from college. In that time, I have witnessed enormous technological advances. The computing power available today in many mobile devices rivals that of some mainframes used back in the early 1970's. Four decades ago, software development was in its infancy. Applications were developed for mostly data processing and scientific uses. Back then a large project was about 50,000 lines of code. By comparison, most 2012 cars have about 300 million lines of code! And I am totally amazed at how young children are using software developed for many of these mobile devices.

The pace of technological advances is increasing so rapidly now that, as observed by inventor and futurist Ray Kurzweil:

"An analysis of the history of technology shows that technological change is exponential, contrary to the common-sense 'intuitive linear' view. So we won't experience 100 years of progress in the 21st century — it will be more like 20,000 years of progress (at today's rate)." [1]

This is illustrated by Moore's Law [1]:



Futurists such as Ray Kurzweil, Bruce Sterling, and Vernor Vinge believe that the exponential improvement described by Moore's law will ultimately lead to what some are calling a **technological singularity** - a period where progress in technology occurs almost instantly.

"The **technological singularity** is the hypothetical future emergence of greater-than-human superintelligence through technological means. Since the capabilities of such intelligence would be difficult for most humans to comprehend, the occurrence of a technological singularity is seen as an intellectual event horizon, beyond which events cannot be predicted or understood.

Proponents expect this to occur sometime in the 21st century, although estimates vary." [2]

A negative implication of **Moore's Law is obsolescence**, that is, as technologies continue to rapidly "improve", these improvements can be significant enough to rapidly render predecessor technologies obsolete. In situations in which security and survivability of hardware and/or data are paramount, or in which resources are limited, rapid obsolescence can pose obstacles to smooth or continued business operations.

Technological obsolescence also has a human side. Technical skills and tools commonly considered "current" as recently as 10 years ago are largely no longer relevant today. Consider the fact that 10 years ago, smartphones and tablets did not exist. Neither did the whole industry of developing small apps for these mobile devices.

Working in the software industry has always required people to continually acquire new skills. However, given the accelerating pace of technological change we're seeing today, people working in the software industry will need to adapt to technology

changes much faster than ever before.

Positive Impact of Rapid Technological Change

The rapid pace of technological change certainly has resulted in many positive changes. For example, we have witnessed dramatic improvements in life expectancy. An average person living during the time of the Roman Empire might have expected to live 25 years. At the turn of the 20th century, an individual had a life expectancy of 50 years. In 2009, the estimated average life expectancy in the US was just over 78 years.

Overall, life expectancy has increased due to 6 major factors – listed below in order of importance:

- Access to clean drinking water
- Universal sanitation
- Significantly improved nutrition, particularly during infancy and childhood
- Vaccination against most common epidemic diseases
- Access to high-quality trauma (accident and emergency) care
- Improved drugs (particularly, antibiotics)

All of these factors have been directly impacted by technological improvements.

Other more mundane examples include widespread access to broadband Internet, which, among other benefits, has enabled more people than ever to work from home. Other technological advancements such as Skype enable people to have virtual meetings without having to travel.

Negative Impact of Rapid Technological Changes

Technological improvement certainly has a down side. Today, people under the age of about 35 can't image life without the Internet, cell phones, texting and e-mail. Many people in the Millennial Generation have never written a letter to someone.

As a result, there have been several negative side-effects that have resulted from all of our technology:



Many people have been killed or injured as a result of texting while driving – a significant problem caused by inappropriate use of new technology.

The advent of remotely piloted drones commonly used in places like Afghanistan has inadvertently resulted in the death and injury of scores of innocent civilians.

And on the lighter side:

People have become way too attached to mobile devices. Several videos have been posted on You Tube of people walking in a zombie-like trance, fixated on their smartphones to the point where they walk into buildings, telephone poles, other people, or even moving traffic.

People can now bring their office with them wherever they are – even when on vacation. As a result, many of these people are over-stressed and can't relax. Being connected to the Internet 24/7 has caused increased stress and anxiety. Some people feel compelled to respond to text messages at all hours of the day and night.

People talk to each other much less – relying instead on non-verbal forms of communication like texting and e-mail. As a result, inter-personal communication has become cryptic and has corrupted our language - at least for some of us. The following is an example of a texting conversation between two teenage girls – Jen and Jill:

Jen: Was up
Jill: Hangin
Jen: Wit who

Jill: Mike
Jen: Omg!! He's fine
Jill: Lol I kno
Jen: brb
Jill: k
Jen: Had to p
Jill: lol thx r u at Mike's house
Jen: Ya
Jill: Omg ur lucky
Jen: y
Jill: duh
Jen: What ru doing 2morow
Jill: idk y
Jen: Lets go 2 da beach
Jill: k
Jen: Mike jus fell
Jill: lmao
Jen: gtg
Jill: k txt me l8tr
Jenn: ttyl

I don't know what I would do if I had teenagers now...

Impact of Rapid Technological Change

Rapid technological change certainly impacts business. Businesses must constantly review and update their work environment, their policies and certainly their products in order to remain relevant in today's highly competitive economy.

For those of us concerned with software quality, we are facing new challenges such as testing software running on several mobile devices and working in agile-like development environments. When technology changes so fast, skills that were once in demand may be no longer valuable. For example, a few years ago, having experience testing in a client-server environment was in demand – not so much today.

While it has always been important to keep your skills current, it has become more and more challenging to do this, given the exponential changes in technology that we are currently experiencing. Also, for some of us, as we get older, it becomes more difficult to adapt to changing technology.

A question we all will have to face at some time in our professional career is **"Have I become technically irrelevant?"**

I've found some indicators that might predict if you are becoming technically irrelevant - regardless of how close you are to retirement.

- You have a stack of technical journals on your desk that you haven't read and are not likely to read in the foreseeable future.
- You haven't learned a new technical skill in a long time.
- Your current skill set was considered current more than 10 years ago.
- You lack many pre-requisites for technical jobs in your field, even your current job.
- You find yourself volunteering for standards committees where the average age of the committee members is over 50.
- You haven't attended a technical training course of any kind in a very long time.
- You find yourself having to Google technical terms more and more often.
- You find it difficult to have a stimulating technical conversation with the youngest members of your project teams.

- People are constantly telling you that things are different today.
- The few technical articles you try to read are way over your head.

If you find that many of these indicators apply to you, you might be in danger of becoming technically irrelevant.

What's it all about, Alfie?

Can you still be a valued asset to your company if, like me, you abhor texting and prefer instead to talk to people face-to-face? Of course you can! What I've learned from 40 years in the software industry is that many businesses continue to make many of the same mistakes over and over again. For all the technological progress we've made in the past four decades, the software industry has still not learned from past mistakes.

- Many organizations view activities like spending time working on requirements as "old-fashioned" and not relevant.
- Many software companies do not use appropriate measures for the software development work they do so they have no way of knowing how good or bad their work and their products are.
- Many project teams are not required to use techniques such as project retrospectives to identify problems and proactively apply lessons learned to future projects.

What I have found in working with over 200 companies is that while today's technology is vastly different, the problems and mistakes I observe haven't changed – they are the same problems and mistakes that companies have made over and over again. I firmly believe in what Roger Pressman once said – using a proven engineering discipline is always appropriate.

Technology companies desperately need people with cutting edge skills. But these same companies also need people who have "been around the block a few times" and can provide guidance in how using "proven engineering disciplines" can address many of their problems.

Pondering the future

Lately I've questioned whether I have become technically irrelevant. Sometimes I feel that I have, especially when working with very young engineers. But many of these young engineers will make the same mistakes that have been made over and over again throughout the history of software development. People who may not have the most up-to-date technical skills can still provide the insights needed to spot impending failure and suggestions for how it can be averted.

Consultants have a saying about some of our clients – many of them have "more money than brains." As long as software companies fail to measure what they do and ignore lessons from past projects, there will be plenty of work for me to do – even if I refuse to text!

'till next time...

Monthly Morsels

Every month in this space, you'll find additional information related to this month's topic.

1. Kurzweil, R., "The Law of Accelerating Returns", March 2001.
2. Kurzweil, R., The Singularity is Near, Penguin Books. 2005.
3. For a list of obsolete skills, visit <http://www.obsoleteskills.com/>

About SQC

Software Quality Consulting provides a full-range of software engineering services for safety-critical industries and mission-critical projects. Our goal is to help create safety-critical and mission-critical software that meets our client's needs, complies with all applicable standards and regulations, with the highest level of quality possible, and in the most cost-effective and timely manner possible.

To learn more about how we can help your organization, [visit my web site](#) or [send me an email](#).

Food for Thought, Predictable Software Development, Act Like a Customer, and ALAC are trademarks of Software Quality Consulting, Inc.
Copyright © 2012, Software Quality Consulting, Inc. All rights reserved.

Graphic design by [Sarah Cole Design](#)