The *Quantum* Age of IT

Why everything you know about IT is about to change

Charles Araujo

itgp™

The Quantum Age of IT

Why everything you know about IT is about to change

CHARLES ARAUJO

IT Governance Publishing

IT Governance Publishing
IT Governance Limited
Unit 3, Clive Court
Bartholomew's Walk
Cambridgeshire Business Park
Ely
Cambridgeshire
CB7 4EA
United Kingdom
www.itgovernance.co.uk

First published in the United Kingdom in 2012
by IT Governance Publishing.
ISBN 978-1-84928-451-6

WHAT OTHERS ARE SAYING ABOUT THIS BOOK

The role of IT has changed in a revolutionary way. Success in this new world order will require a new way of thinking and a new set of competencies. Charles Araujo helps his readers understand the need for change as well as the ways that they must change in order to leverage this exciting opportunity.

Larry Bonfante, CIO USTA; Founder, CIO, Bench Coach, author, *Lessons in IT Transformation*

Araujo's easy-reading style belies the significance of his message. IT has changed fundamentally. Don't get left behind.

Mark Smalley, Ambassador at the not-for-profit ASL BiSL Foundation and freelance IT Management Consultant

Charles has really nailed it for any executive struggling with IT strategy. How IT got here and where it's going. This book really hits at the heart of what is missing in IT today and provides step-by-step approaches loaded with examples for steering the strategic course for IT in the 21st century. An easy read, yet will get the creative juices flowing for any IT professional executive!

Randy Steinberg, Author, *ITIL® Service Operation*, 2011 edition; Principal, Migration Technologies

A bracing indictment of IT dysfunction, and a well-informed discussion of promising avenues for IT improvement. I agree completely with Charlie that a systems approach is essential. Recommended!

Charles T. Betz, *erp4it.com*

What a great, insightful and thought-provoking work! The best part is that Charlie makes it practical and easily understandable with his real-life examples. Finally a road map for IT professionals who want to thrive in the business world into which they are being thrust.

Jaime Rosado Jr., Col. (ret), USAF, Medical Service Corps, FACHE, FAHM

It's courageous to start off a book about the future of IT by saying that IT as we know it is dead. As Charlie is careful to point out, however, IT didn't die overnight; it's been a long time coming. Taking a good look at the history of computing and information technology, Charlie shows us, step by step, why this is happening now, and explains why it is not necessarily a bad thing, and how a new IT business model is needed – and possible. Disruptions of the type we have recently seen in technology-driven businesses have shown that we need to make some hard choices. Those choices are clearly thought through and explained in detail in this book, taking into consideration various approaches to business discipline and to customer focus. Read this. Let's get on with it!

Roy Atkinson, service and support industry analyst and writer

IT has always been a complex journey, with unexpected twists and turns, but no road map. Guess what? Thanks to Charles, we finally have one. I highly recommend this book.

Frank Wander, Founder, IT Excellence Institute; LLC and Former CIO of Guardian Life

The goal of a CIO today is to see the future and enable their organizations to thrive in that new world. *The Quantum Age of IT* provides the insights to make that journey a successful one.

David Hummelberg, Senior VP, Solutions Delivery, the Capital Group Companies

Great! This is the most definitive book on IT transformation I have ever read. It clearly identifies how to perceive, treat, and integrate IT as a true business function, and stop treating it as a specialized staff function, the domain of techies and a money pit for most organizations. It should be required reading for all current and aspiring CIOs, CTOs and CISOs. This paradigm shift is happening right now and this book demystifies the steps needed to make the journey successful. Bravo!!

Roman Hlutkowsky, Former SVP of Human Resources and VP of Operations Technology at FedEx Ground; Principal at the Hlutkowsky Group

With a clear and jargon-free approach, Mr. Araujo provides a compelling vision for a better IT organization and a road map for navigating the fundamental changes that are upon us. Leaders who are ready to forge ahead with a strategy that embraces this brave new world will be energized by these powerful ideas.

Michael Patterson, Ed.D., Adjunct Professor, Pepperdine University Graduate School of Education and Psychology; Co-author, *Have a Nice Conflict: How to Find Success and Satisfaction in the Most Unlikely Places*

This book combines the best of *The Fifth Discipline*, *Who Moved My Cheese?*, *Toyota Kata*, and *Chasing the Rabbit* for both IT executives and business executives alike. As Dr. Deming would say, survival isn't mandatory, so ignore it at your peril.

Gene Kim, founder of Tripwire, author of *The Visible Ops Handbook* and the upcoming books *The DevOps Cookbook* and *When IT Fails: A Business Novel*

EXTRACT FROM FOREWORD

In *The Quantum Age of IT*, Charlie begins with a brilliantly articulated and artfully organized review of the evolution of computing (presently known as IT), leading up to its death. He draws on the past to set the context for his vision of what future IT organizations will look like, comparing and contrasting two different, yet similar, organization models – Strategic Sourcer versus Strategic Innovator – arguing that both are strategic, but emphasizing the importance of picking the right one only after extensive discussions with our business customers.

Building on this foundation, Charlie follows with a description of the five critical traits every IT organization must develop and the five critical skills, or competencies, that all IT professionals need to posses in order to survive and succeed in the new Quantum Age of IT – none of which, by the way, are technical. Each is premised on the diminishing need for deep technology skills, offset by a growing demand for strong business skills. It's clear that the skills that got us here are NOT the skills that will get us to this new age of IT, to quote the book with a similar title from Marshall Goldsmith.

Using numerous examples, including a personal one involving his father, Charlie brings *The Quantum Age of IT* to life. With each chapter, he builds on his basic premise that IT as we know it today is dead, but holds up promise and excitement for what the new IT organization will look like and deliver, drawing on countless experts and noted author's insights to support his thesis.

Charlie concludes his book with a wealth of reference sources, tips and a "How-to" guide for getting started, including information on how to join the Quantum IT Consortium as a way to tap into and collaborate with others who are on a similar journey.

As CEO and founder of the IT Transformation Institute and recognized expert in IT organizational change, this book gives Charlie the broader IT industry exposure he deserves. I am delighted to have played a small role by writing this foreword for his first book, but hopefully not his last.

An IT executive myself with over 25 years' experience, I've come across numerous books on IT that were nothing more than reporting on the latest fad, hypes, or new twists on old themes. *The Quantum Age of IT* isn't one of those; rather, it is a rare gem. To me, it should be on every IT professional's reading list.

Congratulations, Charlie!

Anthony Iorio

Senior Vice President, Information Technology Group

San Clemente, CA

CONTENTS OF FULL VERSION

INTRODUCTION

IT as we know it is dead.

That is a tough way to start out a book, but it must be said. If you do not believe it – or if you are not at least open to considering this as truth – you might just want to put this book down now. Everything that will follow is founded on this belief that everything that we know about IT is about to change – in fact, it already has.

Welcome to the Quantum Age of IT.

"Have you heard about this thing called the Internet?"

It seemed such an innocuous question. Idle chitchat.

"No." I answered. "What is that?"

It was 1990 and I was having an off-hand discussion with a client who worked for a large aerospace firm. Because of her ties to the defense industry, she had very early access to this new technology. She tried to explain it to me, but even though I had grown up in the 'modern' PC era, I could not quite grasp the concept. We chatted for a few more minutes and then moved on. I completely missed the significance of the moment. I did not realize that I had glimpsed the future. That I had seen, for a brief moment, something that would change everything.

Technologies come and go. They always have. How could we know that this one would be the one that changed everything? A lot of very smart people missed it. The

reason is that we could not un-know what we already knew about how IT and technology were supposed to work. We could not escape the bias that this knowledge created. It made it nearly impossible to see the full ramifications of what this technology would bring.

It is called 'the Curse of Knowledge.' It was first illustrated in 1990, the same year as my glimpse into the future, by a Stanford graduate student named Elizabeth Newton.[1] In a series of experiments, she demonstrated that once we know something, that knowledge makes it nearly impossible for us to imagine not knowing it. This affects our ability to communicate and teach ideas because we assume that everyone else must also know what we already know. It also affects our ability to imagine alternatives to our current state. The curse of knowledge becomes a prison of sorts, trapping us in a perspective based on what we know to be true. Until, one day, it isn't true any longer.

It is why the vast majority of innovation during the Information Age came from outside the traditional technology domains. It was driven by people – 'kids,' mostly – who were not inhibited by the trappings of the old truths. They were not subject to the curse of knowledge, so they could imagine new futures that were simply outside the grasp of recognition of those of us who were living in the middle of our current reality. But just because we couldn't see it, that didn't mean that it wasn't happening. The transformation was happening whether we knew it or not – whether we accepted it or not. The new truth had been set in motion back in 1990 (and before) and what was

[1] The term 'the curse of knowledge' was first coined by Robin Hogarth in *The Behavioral Foundations of Economic Theory* (1986).

happening at the dawn of the Information Age was just the manifestation of this new truth.

We are at a similar point in time in IT organizations. The fundamental shifts in technology that began in the 1990s have now led to a fundamental shift in the organizational dynamics of the IT function within enterprises of all sizes. But the same 'curse of knowledge' threatens our ability as IT leaders to see this shift and imagine a fundamentally different future. There is a lot of talk about change and transformation in IT circles today. Most of this is not truly transformational – it is merely incremental. Incremental change will not be enough as we enter this new era for IT organizations.

* * *

This book has three primary goals. ***The first goal is to shake things up***. IT, as we know it, may be dead – but a lot of people do not know it yet. You might be one of them. You go through your day, facing the normal day-to-day challenges. You do your job and you try to do it well. You know that things are changing in IT (when are they not always changing, right?), but you are completely unaware that everything you think you know about your job, your career, your profession – everything – is changing right beneath your feet.

This is not your normal, run-of-the-mill change. This is big. This is game-changing. This is not a flavor of the month. The fundamental business model of the IT organization has changed and you need to understand it. This is so important that the first two parts of this book are dedicated to helping you see what is going on, convincing you that IT really is dead and to understand the IT business models that are rising in its place. You must see this new reality. You must

understand the threat. And you must recognize what these new IT business models mean to your future.

Facing the death of your chosen profession is not a very pleasant thing. But here is the surprise. The death of IT as we know it is really not a bad thing at all. In fact, for those who see this for what it is and seize on it, this represents a tremendous opportunity. But how do you seize it?

The second goal of this book is to lay out the five organizational traits that will define the IT organization of the Quantum Age. In these five organizational traits is both hope and opportunity. These five traits represent the building blocks of the future IT organization. Within them, you will be able to define your future. What is interesting – and challenging – is that these five traits have very little to do with technology. These are not 'capabilities.' This is not about merely obtaining new technical skills. In fact, the term 'trait' is very purposeful. They represent the essential qualities that every IT organization must possess in the Quantum Age.

The five traits that every IT organization must develop and possess are that they must be:

- A Learning Organization
- A Disciplined Organization
- A Transparent Organization
- An Intimate Organization
- A Dynamic Organization.

You may feel that your organization already has some of these traits. You are probably right. That is the good news. Most organizations do. But they are often overshadowed by technical capabilities and are treated as 'soft capabilities'

that are ‘nice to have.’ That is what has changed. These traits are the five characteristics that will define the success of an IT organization in the immediate future. Part III of the book will explain why these traits will now be at the center of how every IT organization operates; it will then break each of them down in detail. The goal is to help you understand these five traits and what they really represent. But knowledge is not enough. You must go in knowing that this understanding will come at a price. Once you understand it, you need to be prepared to act on it.

Which brings us to our third goal. There is nothing worse than knowing that things are changing, knowing that you are in a position to help lead that change, but lacking the skills to do it. Part IV of this book will outline exactly which skills you will need in the Quantum Age. You must be willing to accept that your most valuable assets will not be your technical skills. They will still be needed, but they will not be what provides the greatest value – either to you or to your customers. This part of the book will outline the five specific skills that you need to develop within yourself (and in those around you) to be a force of change, to be relevant to your customers and to take your place in the new IT business model.

Much like the five traits, the five skills that every IT professional needs to master are not technical. This is not about becoming a ‘Cloud master’ (or whatever the current buzzword might be by the time this book is published). But this is also not about a bunch of ‘new age’ soft skills. No meditation is required (although meditation is good!). In fact, the best way to describe these skills is that they are ‘business skills.’ In that, there is good news again. These skills will not be completely unfamiliar – it is just that they are downplayed and not spoken of often in IT organizations

today. But that must change. The five skills of the Quantum IT Professional are:

- IT Financial Management Skills
- Critical Thinking and Analytical Skills
- Communication and Marketing Skills
- Innovation and Collaboration Skills
- Leadership Skills.

These five skills must be developed in abundance. They must permeate every level and every function of the IT organization. They are the foundational building blocks for developing the five organizational traits. And they will feel either foreign or like a complete waste of time to a large chunk of the IT organization. It is for those folks that this part of the book is written. ***The third goal is to help you see that those technical skills that you have developed and honed over all of these years could well be your undoing, unless you are willing to recognize the changes that are occurring***. In these chapters you will see how this fundamental shift in the IT business model has led to a new set of needs – and how these five skills will help you meet them.

* * *

IT as we know it is dead. We are entering the Quantum Age of IT.

It is a time of great hope and opportunity for every IT professional who sees it for what it is. Will you?

Change is never easy. Especially fundamental change like this. It is our natural reaction to hunker down and try to weather the storm. From our experience with the constant

stream of 'flavors-of-the-month' it is easy to take an attitude that 'this too shall pass.' But it won't.

The Quantum Age of IT is upon us. It represents a fundamental shift in everything we know about how IT organizations function and operate. There is no going back. This future will happen. *It is happening*. The only question is whether you will have this future merely 'happen to you' or whether you will lead this change forward.

For those who step up and lead their organizations, their teams and themselves into the Quantum Age, the future is bright. The Quantum Age does not represent a dark era for IT organizations. It represents a righting of the relationship between IT and its customers. It finally creates the relationship between IT and its customers that should always have existed. It ushers in an era of explosive growth in the application of technology to solve real and meaningful business problems – and to drive ever-increasing value for our customers. It offers an opportunity to make IT fun once more. A chance to finally move past the mundane to the strategic.

The journey into the Quantum Age of IT will be hard. There is no getting around that. A lot will be asked of you. You will be asked to learn new skills and to fundamentally change how you operate – and even how you think about your role.

But the journey will be worth it. At the end of this road is the IT organization that our customers have always wanted – the IT organization that we always wanted. One that is a fun place to work. One that provides consistent, meaningful and measurable value to your customers every day. One that is an engine for business growth and profit. One that not only can rapidly adapt to changing business needs, but

also can help create competitive advantage. And one that does it all seamlessly and transparently, allowing our customers to focus on their business challenges and opportunities knowing and trusting that we are there with them every step of the way.

Welcome to the Quantum Age of IT.

CHAPTER 1: THE HISTORY OF OUR DEATH (WHY THE MODERN IT STRUCTURE HAS FAILED US)

Jeff Winston was on the phone with his wife when he died.

First line from *Replay* by Ken Grimwood

The book *Replay* by Ken Grimwood is one of my favorite books of all time. It was a bit of a sci-fi cult classic when it was published in 1986. It was at once entertaining and profound. It tells the story of a man who dies suddenly at the age of 43 – only to wake up back in his freshman year of college. He learns that he has been given a great gift. A chance to live his adult life again. A 'do-over.' He decides that he will not make the same mistakes twice and lives his life differently. Until he reaches age 43 – and he dies again.

As this cycle repeats, he comes to a realization. He learns that changing his past is not the road to changing his future. He finds that his past experiences were a part of who he was and that spending his life looking backwards was only squandering the one thing of value that he really had – his future.

"The IT organization was in the middle of its next reorganization when it died."

Perhaps that should have been the opening line of this book. Much like Jeff Winston, we are at a similar point in the life of the modern IT organization. (As a happy coincidence, the modern IT organization is about 45 years old!). Our organizations have grown and evolved – in many cases, without much conscious thought. There was always too much work to be done to be contemplative. Sure, some

strategic planning took place and there have always been the pundits and the prognosticators, but for the most part IT leaders were far too busy getting things done to waste time imagining their future. And, for the most part, it worked out just fine.

Then we died.

We just didn't know it.

But as Jeff Winston realized, this death is an amazing gift. It is an opportunity to give a fresh, new life to the organization. The lesson that Jeff Winston learned is the same one that we must now take to heart. There is nothing to be gained by complaining about our past or living in a world of 'what-ifs.' Our future lies in front of us, not behind us. But there are lessons for us in our past. There is clarity for our future to be found in the things that led us here. By understanding our past, we can better accept our today and then guide our tomorrow with an eye toward the future that we want to create. In order to envision our future, we must begin with the past.

The history of our death – part 1

How the function of IT came to be and the evolution of our organizational structure

The first computers were not computers at all.

The term 'computer' dates to the mid-18th century and literally referred to mathematicians whose job it was to perform long and arduous calculations by hand. They were typically hired by scientists to speed what would otherwise be a laborious process. Over time, the 'computers' realized

the benefits of dividing their tasks and creating specialization. Eventually they created large books of 'premade' tables of already completed calculations so that greater calculations could be built from them. The first electronic computers were essentially created to replicate and replace the manual process that 'human computers' were performing. That fundamental process has continued to be the foundational drive behind all computing. To take what humans do slowly and imperfectly and enable it to be done rapidly and accurately.

By the time the modern mainframe computer was created in 1951,[2] this simple vision had spawned an entire industry, its own scientific discipline and, most importantly for our purposes, the beginnings of a new profession. By 1964, it was clear that there was a huge market for computers. But the complexity and cost of the technology made it difficult for most organizations to make the leap. It was into this market that IBM introduced the computer that would largely define the industry going forward, the IBM System/360 Series. It was a compatible series of computers that were all capable of running the same software. Based on this common architecture, it opened up vast possibilities for customers. The fact that it fit into IBM's existing infrastructure, combined with IBM's legendary sales force, suddenly made it practical and affordable for companies to begin purchasing the IBM System/360 Series and utilizing them for a wide range of purposes. As company executives began using their new technological marvels, however, they soon realized that they needed to employ a staff of people

[2] The UNIVAC 1 was developed in 1951 and delivered to the US Census Bureau in 1952.

who could program and operate them. And the function of IT was born.[3]

Technical foundations

From the very beginning, computers were set apart from 'normal' life. They were born in one of the greatest eras of technological advancement the world had ever seen. During the fifty years preceding the dawn of the commercial mainframe, we had been introduced to mass-produced cars, commercial air travel, and vast levels of 'automation' on both industrial and consumer levels. Everything from the automated assembly line to dishwashers, washing machines, and, of course, television had come on the scene in the short fifty-year period before the introduction of the commercial computer.

The world was in awe of technology. In 1955, Walt Disney inspired imaginations around the world with his new Tomorrowland area of Disneyland. In the 1950s and early 1960s there were over 150 movies released that dealt with the wonder of the modern era and imagined wild futures of flying cars and robots. It was into this world that the computer began its journey into the mainstream. It is no wonder that computers and the folks that operated them were viewed as something separate from the rest of the company.

The very first computers required highly technical people to design and implement them. They were advanced mathematicians and technicians who built and managed the entire platform. While the great innovation of the modern

[3] At the time, the function was often referred to as data processing.

mainframe computer was that it was 'programmable,' it still required a very technical skill set to write the binary code necessary to make it work. The work of writing this code was often long, arduous, and fraught with error. It was easy to make a simple mistake in the sometimes millions of lines of binary.

Companies, however, began to see the promise. They began imagining more diverse and more complex tasks that computers could handle. What started as a machine to do 'computations' was suddenly being used for a wide variety of purposes. With each new use imagined, the challenge of programming it became more acute.

Because of this complexity, two things happened. First, it became clear to organizations that the people that they needed to program and operate these new computers were going to be a special breed of people. This was not going to be something that just anyone could do. They would need to hire or train people with this specific skill set.

Second, the computer companies realized that they needed to do something. It was becoming apparent that, in some cases, it was taking longer to write the program to automate a task than it would have taken simply to do the task manually. So, they began developing 'programming languages' that made the job of programming a computer much easier. Languages such as FORTRAN and COBOL were introduced and represented the first major shift in how computing was done.

Specialization and separation

The creation of the first programming languages created a fundamental shift in how computers were used and

operated. They opened up a world of possibilities for organizations by making it easier to do more complex and specialized tasks. This created an explosion in their use and was a boon for the growing computer industry. Suddenly, there was intense demand for programmers who could harness the power of these new investments.

The programming languages ushered another new aspect into the world of computing – specialization. Up to this point, computing was essentially a unidimensional discipline. The advent of programming languages and the large number of new computer companies that arose during this era brought with them a large number of subdisciplines specializing in specific platforms, programming languages, or industries. It was no longer enough just to be a programmer. Companies were looking for a "COBOL programmer on the DEC platform with experience programming financial systems."

This first level of specialization began creating divides. While at the beginning it was common for people to learn both FORTRAN and COBOL, over time people began to self-segregate. Scientifically oriented organizations were most interested in using their computers for complex calculations. The programmers were, therefore, predominately focused on FORTRAN because of its more advanced calculation capabilities. Business-oriented organizations concentrated on automating workflows and less complex calculations, so they focused on COBOL, which had been built to specifically meet this need. It became clear that there was not a great deal of crossover and so programmers began 'picking sides.'

This was not adversarial. Overall, IT people have always been collegial. It was more like the Tower of Babel. In the

beginning we all spoke the same language. We could communicate, share stories, and trade roles. But over time, we began to forget. As programmers picked their sides and became specialized, they had little to no need for the other languages. So, we ended up working in different domains, speaking our different languages and working on different problems. Even within the same company this happened. If a company had a need for both technical computation and business-oriented computing, the two programming teams would self-segregate, each working on their own problems.

Soon, specialization became separation. Entirely separate camps of programming disciplines developed. They often involved different approaches, methodologies and documentation standards. The separation continued with the proliferation of additional computer makers, with each introducing its own separate set of parameters. What had begun as a singular approach to programming had evolved into a wide range of programming disciplines, each demanding different skills – and often different perspectives on how things should be done.

It was the first of many cultural divides to come.

The first silos

While programming skills were being internalized and stratified to meet the specific and increasingly unique needs of organizations, a separate discipline was developing elsewhere in the world of computing: the *computer operator*.

Originally, computers were operated in much the same way that the card tabulators had been operated before them. ‘Programs’ came in the form of punch cards and simply

needed to be loaded in order to run the computer. The job was simple and was typically done by the same people that had put the old punch cards into the mechanical tabulators.

As computers became more intricate, however, this broke down. As they moved from punch cards to tape and from binary to programming languages, it became clear that the old way of operating the computer was not going to work. Companies realized that they had to hire people who could function in this new and specialized world and keep everything running. The role of the computer operator was born.

The division of labor was pretty simple and straightforward. The programmers wrote the programs and the operators operated the machines. They were fairly distinct disciplines and, while it may seem strange to us today, they had little dependency on one another. There was really nothing that operators could do to impact the programmers. As long as they ran the job, switched the tapes or whatever other operational task was required, whether or not the program ran correctly was solely in the hands of the programmer.

The same was largely true from the other direction. The operators did not much care what was in the program. There was little that the programmers could do that would have a major impact on the operators' ability to do the job. As long as they had sufficient instructions on the operational tasks required, the worst thing that could happen from their perspective was that the job would not run.

The discipline of computer operations was task-oriented. It was about executing a series of tasks consistently, reliably, and predictably. It was important work. But it was not one that required a high degree of creativity or ingenuity. A

different type of person was attracted to the role of a computer operator. They were more mechanical. They were more task-oriented and enjoyed the idea that they were the guardians of the kingdom, making sure that everything worked as it was designed.

They also saw themselves as separate from the programmers. The divide continued.

The kingdom grows

Over the ensuing decades, this fundamental structure grew and expanded. New languages were added, new technologies were deployed, but everything ended up in one of these two 'silos.' In part, this was because the fundamental technology structure had not changed much. Improvements were made to the capabilities of specific components of the technology, making things faster or more efficient, but the basic architecture did not change.

Eventually, things like Project Management Offices (PMOs), security teams, and other ancillary functions were added to the IT operation. Sometimes this resulted in creating separate subfunctions within IT, but for the most part the two primary silos persisted.

After a time, these silos became fully entrenched. As IT grew from a technical discipline into a career, IT managers began to have a vested interest in this structure. Inertia set in. People were happy to keep things the way they were. After all, they seemed to work, right?

At no point did anyone really stop and ask if this was the right approach. IT and 'computing' were never seen as a strategic core competence during the early years. Viewed as

largely technical functions, they did not warrant much strategic energy. They simply evolved.

The challenge was that it evolved based on a set of basic assumptions regarding the relationship between Applications and Infrastructure and between IT and its customers, who would themselves evolve over time, eventually leading to a fundamental break in these relationships.

The history of our death – part 2

The rise of the IT corporate culture and why we work the way we work

You can't win. You know that, don't you? It doesn't matter if you whip us, you'll still be where you were before, at the bottom. And we'll still be the lucky ones at the top with all the breaks. It doesn't matter. Greasers will still be Greasers and Socs will still be Socs. It doesn't matter.

From *The Outsiders*, by S.E. Hinton

The book and movie *The Outsiders* tells the story of two rival groups of teenagers locked in an age-old battle of cultural warfare. They come from different sides of the railroad tracks that physically and metaphorically divide those who have from those who have not.

The Socs (pronounced 'so-shus') come from educated and prosperous families. They are the 'thinkers.' They do not have to do manual labor. They will go off to college and, one day, they will be the boss. They see themselves as better than the rest.

The Greasers are working-class. They get their hands dirty. They are the ones that have to do the real work. They look at the Socs with disdain, living in their little world, oblivious to the reality of how things really get done. They see themselves as 'real' and the Socs as 'fakes' and pretenders who are blinded by fortune.

The separation between these two is clear. It is not just that they come from different backgrounds. They see themselves as different from one another. They cannot imagine that they have anything in common with one another. They each see their view as the right one – as the one that must be defended and protected.

The evolution of IT has led to our own version of the 'Socs' and the 'Greasers.' While there are no 'rumbles' going on in your typical IT organization, the cultural separation that has developed over the years is palpable. It may not be openly adversarial, but in most IT organizations the divide in worldviews and perspectives between 'Applications' and 'Infrastructure' is real.

It is not difficult to see why this occurred. But it is instructive to understand how this divide between the 'Socs' and 'Greasers' of IT developed and how it will impact IT organizations as the fundamental business model of IT shifts.

Different skills and different worldviews

The cultural wars that are playing out across IT organizations today started at the very beginning. The early computers were incredibly complex to design and build, but operating them was not.

Operating a computer involved loading some cards or loading a tape or invoking a few commands. Operators would monitor systems for jobs that did not run properly and either let the programmers know that they had failed or restart them if the error seemed environmental. Computer operation was mostly a manual function that required repeating many tasks over and over again.

Programming, on the other hand, was a sophisticated and nuanced activity that required a highly specialized skill set. It was the programmers who were the masters of the domain. Operators were the hired hands. Programmers were the translators between the business need and the code that would perform the needed function. They were equal parts architect and detective, designing and troubleshooting previously unimagined pieces of software.

As a result of these different skill set requirements, you could hire an unskilled person and train them to become an operator. That was much more difficult to do for programmers. A natural cultural divide, therefore, developed between the operators and the programmers. These were not equal, but separate, roles. But this was not as adversarial as it might sound or seem today. Both sides understood and accepted their role. If you were an operator, there was no shame in it. You had a job to do and you took pride in it.

It was really not much different to the relationship that existed during the dawn of the industrial age. There were engineers who designed and built and there were the machine operators in a factory who produced. It was the way things worked and everyone understood their role. This model worked in IT organizations for many years. The

cultural divide was real, but it was not a source of stress. It just was.

Until, that is, something changed.

The rise of distributed computing changes the rules

The rise of distributed computing in the late 1970s and early 1980s changed the paradigm. The computer was no longer just a big box sitting in the basement that merely had to be operated. Suddenly, there was a 'network' of systems that had to be connected together in order to produce the desired result.

Up to this point, operators had just 'operated.' But as complexity fed into both specialization and separation, new operating platforms made up of multiple different systems had to be designed. It was no longer just operations. Creativity was required to design a system that comprised these various parts. While the role of operator would continue to exist, entirely new roles were being born to handle these new responsibilities.

As the complexity of IT operations expanded, new management and engineering roles were created. Unlike the simple operational roles of the past, these roles required much higher levels of skill and education and much greater interaction with the programmers. Systems were no longer islands unto themselves. Programmers were beginning to write programs that depended on environments outside their primary system platforms. This meant that the fundamental relationship in which programmers 'programmed' and operators 'operated' was breaking down. It was no longer that simple. Developing and executing an effective

application now required that the two disciplines work together to design a system that would actually function.

There was only one problem: the cultural biases that had been formed at the beginning persisted. What had once been a simple paradigm was now fracturing. This began to create a significant amount of stress in the relationship between the programmers and what was now beginning to be called 'IT operations.'

Like the 'Socs' and the 'Greasers,' the two sides did not see eye-to-eye. The application teams continued to see themselves as the masters of the domain and did not fully understand the growing complexity of the infrastructure on which their applications ran. IT operations professionals, on the other hand, became frustrated that the programming teams did not give them credit for the work they did and for the complexity of their task, but also failed to realize the growing complexity that the application teams were dealing with at their end.

A future hindered by our past

As distributed computing rose and then gave way to the increasingly distributed and complex approaches that would follow, this complexity changed the fundamental relationship between the programmers (now mostly called 'application development') and the operators (now mostly called 'infrastructure operations').

The simple and clean model that had guided the early relationship – programmers 'program' and operators 'operate' – was woefully obsolete. Development could no longer be done in a vacuum. Server architecture, database

design, and network architectures all could have a dramatic impact on application performance and even functionality.

In addition, as the applications themselves began to rely on increasing integration with other applications, the challenges multiplied. In many large organizations, it became common for no one in the organization to truly understand the full end-to-end design of a complex system.

> It's different [from other engineering disciplines] in that we take on novel tasks every time. The number of times [civil engineers] make mistakes is very small. And at first you think, what's wrong with us? It's because it's like we're building the first skyscraper every time.
>
> Bill Gates (Microsoft, 1992), from the documentary *The Machine That Changed the World*

This quote from Bill Gates was still from the early stages of distributed computing and before the explosion of the modern, Internet-driven economy that we know today. The situation has only become more complex. Every day, IT organizations are creating ever more complex systems that require intricate levels of integration to function properly. They span the disciplines of Infrastructure and Applications, but are hindered by the cultural artifacts of our roots.

The problem here is that the fundamental complexity of the systems demands a cultural posture of collaboration and mutual design. Application and Infrastructure teams, however, remain stubbornly rooted to their traditional worldviews. The 'Application Socs' do not want to be bothered with the infrastructure elements that may impact their applications. They want to be left to do their coding in peace. The 'Infrastructure Greasers' want only to remain

focused on their technical domains, leaving business interactions and understanding to the application teams.

This is why to this day you will hear infrastructure teams gripe that they need ‘dedicated windows’ to complete their infrastructure changes – without having to coordinate with the hundreds of application teams that may be impacted. It is why you will hear application teams complain that the infrastructure guys should not be questioning or challenging anything about how the application is architected or developed – they should just stick to operations.

The Socs and the Greasers are alive and well. We are a victim of our past. We are a product of our roots. And while many have complained about it, it probably could have stayed this way forever. Except that something fundamental changed in 1999. Although no one realized it at the time, it would eventually lay bare the dysfunction of the internal IT organization and set the stage for the Quantum Age of IT.

Prelude to a change

Up to this point, there had been a constant evolution of technology, but the fundamental underpinnings of the way IT worked had not really changed. As new technologies had been adopted, new silos were created, but the fundamental building blocks of the organization, both structural and cultural, remained basically consistent. The cultural divide between Applications and Infrastructure had continued and grown and had pushed some IT organizations to the brink, but there was still little pressure to change the way IT organizations actually worked. Everyone complained about this cultural divide, but it was just the way the IT organization operated. The way it always had.

Then, in 1999, Google burst onto the scene.

Most did not recognize it for what it was. But it was the prelude to the fundamental shifts that were about to rapidly descend on IT organizations everywhere. It would take over ten years to fully materialize, but it would fundamentally change perceptions of how technology was supposed to work.

As people discovered Google, they discovered a power they had not known before. Suddenly, a world of information was open to them. All with the simplicity of a single search box.

Chapter 1 key points

IT as we know it is dead. It has happened gradually and right before our eyes. But this is not a bad thing. On the contrary, it represents a great opportunity. A gift. But only if we recognize it for what it is.

Here are the key points you should remember from this chapter:

- The IT function was never "created." It simply evolved as the new technology of computers burst onto the scene.
- As we evolved, we began separating ourselves into natural silos based on specialization (programmers versus operators) and platforms (e.g. COBOL versus Fortran).
- This eventually created a divided IT culture and a sort of Tower of Babel where entire teams within the IT organization no longer spoke the same 'language.'
- But as computing evolved and became more complex, the basis for this cultural divide broke down, fractured

and became a source of great frustration where the cultural norms that existed were in conflict with reality.

- While most recognized these cultural divides and silos, it might have remained unchanged except that something happened that would change everything.

--- END OF EXTRACT ---

To find out more about this book or purchase the full version, please visit one of the following websites:

www.itgovernance.co.uk/products/3941

www.itgovernanceusa.com/product/2598.aspx

www.itgovernance.eu/p-929.aspx

.