Cloud is not Rocket Science but it is Computer Science

Lori MacVittie, 2010-30-08

That doesn't mean it isn't hard - it means it's a different kind of hard.

> Computer science is no more about computers than astronomy is about telescopes
>
> E. W. Dijkstra

For many folks in IT it is likely you might find in their home a wall on which you can find hanging a diploma. It might be a BA, it might be a BS, and you might even find one (or two) "Master of Science" as well.

Now interestingly enough, none of the diplomas indicate anything other than the level of education (Bachelor or Master) and the type (Arts or Science). But we all majored in *something*, and for many of the people who end up in IT that *something* was Computer Science.

There was not, after all, an option to earn a "MS of Application Development" or a "BS of Devops". While many higher education institutions offer students the opportunity to emphasize in a particular sub-field of Computer Science, that's not what the final degree is in. It's almost always a derivation of Computer Science.

Yet when someone asks – anyone, regardless of technological competency – you what you do, you don't reply "I'm a computer scientist." You reply "I'm a sysadmin" or "I'm a network architect" or "I'm a Technical Marketing Manager" (which in the technological mecca of the midwest that is Green Bay gets some very confused expressions in response). We don't describe ourselves as "computer scientists" even though by education that's ***what we are***. And what we practice is, no matter what our focus is at the moment, computer science. The scripts, the languages, the compilers, the technology – they're just tools. They're a means to an end.

CLOUD is COMPUTER SCIENCE

The definition of computer science includes the word "computer" as a means to limit the field of study. It is not intended to limit the field to a focus on computers and ultimately we should probably call it **computing** science because that would free us from the artificial focus on specific computing components.

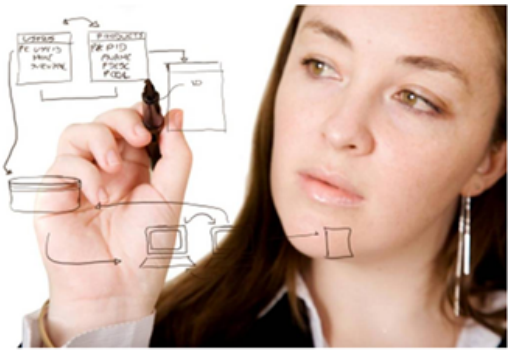
> **Computer science** or **computing science** (sometimes abbreviated **CS**) is the study of the theoretical foundations of information and computation, and of practical techniques for their implementation and application in computer systems. It is frequently described as the **systematic study of algorithmic processes that create, describe, and transform information**. [emphasis added]
>
> -- Wikipedia, "Computer Science"

Interestingly enough, Christofer Hoff recently made this same observation in perhaps a more roundabout but absolutely valid arrangement of words:

> Cloud is only rocket science if you're NASA and using the Cloud for rocket science. Else, for the rest of us, it's an awesome platform upon which we leverage various opportunities to improve the way in which **we think about and implement the practices and technology** needed to secure the things that matter most to us. [emphasis added]
>
> /Hoff -- Hoff's 5 Rules Of Cloud Security…

Hoff is speaking specifically to security, but you could just as easily replace "secure" with "deliver" or "integrate" or "automate". It's not about the platform, it's the way in which we ***leverage and think about and implement*** solutions. It is, at its core, about an ***architecture;*** a way of manipulating ***data*** and delivering it to a person so that it can become ***information***. It's ***computing science***, the way in which we combine and apply compute resources – whether network or storage or server – to solve a particular (business) problem.

That is, in a nutshell, the core of what cloud computing really is. It's "computer science" with an focus on architecting a ***system*** by which the computing resources necessary to secure, optimize, and deliver applications can be achieved most efficiently.



COMPUTER SCIENCE is about SYSTEMS, ARCHITECTURE, CONNECTIONS, and RELATIONSHIPS

COMPONENTS != CLOUD

Virtualization, load balancing, and server time-sharing are not original concepts. Nor are the myriad infrastructure components that make up the network and application delivery network and the storage network. Even most of the challenges are not really "new", they're just instantiations of existing challenges (integration, configuration management, automation, and IP address management) that are made larger and more complex by the sheer volume of systems being virtualized, connected, and networked.

What is new are the systems and architectures that tie these disparate technologies together to form a cohesive operating environment in which self-service IT is a possibility and the costs of managing all those components are much reduced. Cloud is about the infrastructure and how the rest of the infrastructure and applications collaborate, integrate, and interact with the ecosystem in order to deliver applications that are available, fast, secure, efficient, and affordable.

The cost efficiency of cloud comes from its multi-tenant model – sharing resources. The operational efficiency, however, comes from the integration and collaborative nature of its underlying infrastructure. It is the operational aspects of cloud computing that make self-service IT possible, that enable a point-and-click provisioning of services to be possible. That infrastructure is comprised of common components that are not new or unfamiliar to IT, but the way in which it interacts and collaborates with its downstream and upstream components ***is*** new. That's the secret sauce, the computer science of cloud.

WHY is THIS IMPORTANT to REMEMBER

It is easy to forget that the networks and application architectures that make up a data center or a cloud are founded upon the basics we learned from computer science. We talk about things like "load balancing algorithms" and choosing the best one to meet business or technical needs, but we don't really consider what that means that the configuration decisions we're making are ultimately making a choice between well-known and broadly applicable algorithms some of which carry very real availability and performance implications. When we try to automate capacity planning (elastic scalability) we're really talking about codifying a decision problem in algorithmic fashion.

We may not necessarily use formal statements and proofs to explain the choices for one algorithm or another, or the choice to design a system this way instead of that, but that formality and the analysis of our choices is something that's been going on, albeit perhaps subconsciously.

The phrase "it isn't rocket science" is generally used to imply that "it" isn't difficult or requiring of special skills. Cloud is not rocket science, but it ***is*** computer science, and it will be necessary to dive back into some of the core concepts associated with computer science in order to design the core systems and architectures that as a whole are called "cloud". We (meaning you) are going to have to make some decisions, and many of them will be impacted – whether consciously or not – by the core foundational concepts of computer science. Recognizing this can do a lot to avoid the headaches of trying to solve problems that are, well, unsolvable and point you in the direction of existing solutions that will serve well as you continue down the path to dynamic infrastructure maturity, a.k.a. cloud.

Related Posts

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com