# Challenges of Technology Evangelism

How do you overcome the challenges of Technology Evangelism when your barely keeping your head above water just like everyone else?

# Predictions

When I was younger I told my mom every book, song, and movie would be online. She thought I was nuts.

When I was 12 and playing with my TRS-80 I imagined if I could draw a dot on the screen in all the right places I should be able to make an image that looked real. I was never able to achieve this because I ran out of memory but years later we do this without a second thought.

In 2005 I implemented a deployment strategy that delivered on a regular cadence. At the time it seemed that users could more easily predict when their applications would change, developers could more easily predict which branch their code would be completed in, and delivering smaller changes more often would reduce our risk. Many thought I was crazy but it worked and it worked well. Our quality improved. Nearly 10 years later I’m still promoting the practice but at first, again, I was thought to be crazy. Finally I discovered DevOps and Continuous Delivery. It’s exactly what I was previously suggesting and implementing but now that it’s a well-known movement only half of my peers think I’m crazy. The other half know it.

I spoke with a programmer who was very skilled at his craft and OOP centric. I suggested that OOP had run its course and that we really only needed 4 or 5 design patters of the 23 design patterns promoted by the “Gang of Four”.

In the early 90’s I was inspired to become a programmer because of the 3D UI worlds of Jurassic Park, Star Trek. I quickly realized it was beyond my skills but wondered why others weren’t pursuing it. I imagined that it must be possible to display more useful information to the user with a 3D environment than with a 2D environment that mimicked paper forms. Now it’s obvious that we are only 2 or 3 years away from such interfaces being common place with the advent of Hololens and Oculus. Still, my peers think I’m crazy to suggest we start learning this technology now so we are prepared for it when the opportunity presents itself.

When the IoT became finally possible I began socializing those in my circle of influence. My goal was to let people know what was possible so their creativity might have a focus. At first the term “Internet of Things” was literally laughed at. I think there is a certain price we pay for “not” being ignorant and a certain pass we must give to those who are. It’s not a fair position but it is the correct position if we are going to move technology forward.

In 2010 when HTML5 started to hit the scene I realized immediately that this would replace all other applications and there would be no need to write separate code bases for each platform. I was met with much skepticism and at one point called out, publically, as “completely narrow minded” because I thought one code base could run everywhere. Today writing one code base for all devices is a reality (hybrid apps and responsive design) but between 2010 and 2015 we’ve written a lot of applications. Many of these need a re-write, especially the Silverlight apps. Oh, and who is narrow minded in that situation?

We need to start learning and understanding how to deal with the cloud now so we are ready to move to the cloud when it’s obviously the right solution. In a few years on premises will become a thing of the past. Even our precious data will be considered safer in the cloud than it is in our own data centers. Our own data centers will be considered no safer than our home network. One day the board of our company will ask where our data is. When we say it’s in our data center the response will be that of disgust, shock, and a realization we have not kept up with the times and as a result we have put corporate data at risk. Somebodies getting fired.

# Cost of not being ready

I hate to even imagine the cost.

**Billions lost (“I could have been a contender!”)**

If my mother had listened to me and invested in my crazy ideas when I was 12… I would have started google but it would have been called booble.

**Brittle Software (confidence lost)**

If I weren’t met with such resistance on Continuous Deployment, even before it had a name, many line-of-business applications would not be so brittle today.

**Opportunity Cost (IT reputation lost)**

Having built applications that target a hardware platform has limited the reach of some applications. Missing the “Hybrid-app” boat is going to be expensive.

**Missed Opportunities (untold millions lost)**

IoT and 3D UI(s) seem like a thing of the future but if you don’t prepare not then they will be things of a much more distant future for you. Those of us in manufacturing industries are missing out on untold millions because our IT guys lack the imagination to see where IoT can reduce down time and predict when machines require maintenance before they break. What is probably a greater sin than lack of creativity is stifling the technology evangelist who does see the potential.

# It’s hard

So yes, being a technical evangelist is hard. Doing the hard work of educating yourself, spending many long hours of your own time reading and tinkering and imagining where the next evolution of technology is coming and then only to have that information greeted with skepticism and snarkyness is hard. The right thing to do is suck it up and pretend it doesn’t matter. But for a moment, just one moment, I think we can all be honest. Every now and then…

Yes it’s hard. Yes, the closed minded are all around us. They don’t even know they are closed minded.

That’s ok. It’s just part of the job.