

***Good Project Gone Bad: Planning, Managing and Delivering Complex Technology Projects***  
**Nik Honeysett – Chair, AAM Media and Technology SPC**

As a recovering Museum Technologist, this session handout tries to capture some 'gems of wisdom' that I've learned over the last 20 years of my involvement in museum technology projects, which have been for the most part, public-facing projects. I'll frame these as principles. Some of these principles will apply broadly to technology projects and some will focus on specific types of technology projects. These principles are not going to teach you how to magically have a totally successful technology project, but I hope they will help you think differently or offer you a different perspective that will contribute to a successful next project, or even a current one.

In an ideal world, all your technology projects will be uneventful but successful. Sadly, that is not the nature of technology, as I'll describe, technology is a process not a thing. If you have a project going bad, your goal is resolution. Your ability to resolve it is going to be based on the decisions you make before and during the project. First a quote...

*If you put tomfoolery into a computer, nothing comes out of it but tomfoolery. But this tomfoolery, having passed through a very expensive machine, is somehow ennobled and no-one dares criticize it.*

- Pierre Marie Gallois

And now, those principles...

**COMPLEXITY** - What does it mean to be a complex technology and why is that important? If you're a technologist, you know – complex software, hardware, interfaces, etc. But more than likely, the two most important people in any public-facing technology project are non-technologists: The museum director and the public. Sadly, neither cares whether it's a complex technology or not, they just want it to work, flawlessly. In the case of the museum director however, he or she needs to understand the nature of a complex technology because a complex system with many moving parts and interfaces is statistically more likely to fail than a simple system.

So, let me elaborate this principle with something more familiar and explain why it is important for a museum director to understand the nature of a complex technology. Take the example of an audio guide which is a simple system that, because it has everything it needs to carry out its function, goes through a full and comprehensive test when you turn it on and hand it to a visitor. The reliability of that device sets a precedent for any replacement system such as a more complex handheld device that, because it is more complex, is going to be statistically less reliable.

**DESIGN FOR FAILURE** – I'm going to make a distinction here between designing a system to work rather than designing it to fail. When planning a technology project, we tend not to discuss the concept of failure, we build systems with the belief that everything will work as intended. This is an unrealistic expectation. When an escalator fails, what does it become? Stairs. No loss in functionality. A technology system should be designed similarly. There's a similar concept in programming called 'exception handling'. It is what to do if an unexpected event or problem occurs, the goal being to handle it gracefully with minimal disruption to service.

**TRUST** – So, we have a technology and it broke. If you're a technologist, you understand that technology is a process, you debug, rebuild, relaunch. As a technologist, the fact that the technology broke actually gives us more insight into how it works. If you're a non-technologist, you likely think of technology as a 'thing', and it broke. A museum director who is sponsoring a public-facing technology that fails, immediately loses faith in the people who put it together. Why should they trust them to be able to fix it?

Who is responsible? Is the technologist responsible for not being skilled enough to deliver a working product? Is the director responsible for not giving the technologist a second chance to fix it? The relationship between museum director and technologist should be one of trust, one that enables during the planning phase, the discussion of the risks and potential failures of a complex technology project. The museum director, as sponsor, should be aware of every possible failure and what the contingency is. The last thing you want to happen is for the sponsor to enter the seven stages of grief starting with 'Shock & Disbelief'.

**THE MAN ON THE CLAPHAM OMNIBUS** – is a legal definition (in the UK) of a reasonably educated and intelligent but non-specialist person — a reasonable man. The technologist's job is to explain technology in a way you, as a museum director, can understand. At any level, if you are a non-technologist, be wary of hiring a technology person who cannot do this. The more complex the project, the more important this is. This is a similar concept to decision making in a changing technology landscape, one of Google's Ten Golden Rules: When a reasonable person can't understand it, don't do it.

**MUTUAL GOALS** – In a complex project that requires multiple vendors or teams to contribute their piece of the puzzle, you need to ensure that their success matches yours. For example, if vendor 1 delivers widget 1 and vendor 2 delivers widget 2, and the project requires that widget 1 and 2 work together, their success should not be to deliver their widget to you, it should be that the completed project works. You should police this through your contract.

**CONTRACTS** – A contract is a good thing that should be embraced. It's a way of ensuring that everybody has a good experience and gets what they want out of a project: The vendors wants the work, you want a successful project – use the contract as a guarantee for both sides. Some ideas: A payment schedule that matches deliverables and goals; penalties on both the museum and vendor side that ensures each is providing what they agreed, when they agreed; if you want specific people to work on your project, name them; describing what both parties will do in the event of failure; describe how you will handle any disputes that arise.

**LAUNCH PARTY** – A technology project is not an excuse for a launch party. Don't focus the project's attention on a single point in time. A technology project is an operational investment, it's a program of ongoing work. Focusing on the launch is like focusing on the birth of a baby, but that's when the real work starts. What does your project look like in a year's time? Two year's time? Five year's time?

**STEINBACH PRINCIPLE** – Named after Len Steinbach who, much more eloquently, described one of Google's Ten Golden Rules. Google says: Make data-driven decisions based on quantitative analysis. Len says: *...there are so many desirable ways of using technology... your choice must not rely on intuitive, personal, parochial or political reasons...*

**FOXWORTHY PRINCIPLE** – Your project might be going bad if... You have missed deadlines; project meetings keep getting postponed; project staff are constantly making excuses; project staff start avoiding you; project staff start having excessive time off or leave; project staff are constantly having late nights or working weekends.

**HOLLYWOOD ENDING** – Why is it that Hollywood movie ending's are so bad? Well, there are some parallels to technology projects. Your project will go bad if it is... under funded; under supported; under scrutinized; under staffed; and last but not least, you move the goalposts: like deliverables, scope and hardware technologies.

**RESOLUTION** – If you have a good technology project going bad, your primary objective is resolution: what are the decisions you have to make and what are the things you need to do to resolve the situation? Take a pause, analyze the situation and enter a resolution stage. Your three options are, Fix, Mitigate, Start Over: If you are a technologist, a primary objective should be to restore confidence in you, your skills and the technology.

**FLEXIBILITY** – To be scalable and reusable, technology and data need to be designed in a modular way. This modularity ranges in concept from the three-tiered application architecture, separating data from logic from presentation, to the concept of fielded data. Any public-facing system that sees the presentation and logic layers as discrete and separate from the content, is in a much better position to recover from problems. Building a modular system provides the ability to make changes to one layer without affecting the others, it also provides an inbuilt protection framework that limits issues and problems to a single layer

**PRINCIPLE OF ENDING WITH A QUOTE** - *There is no safety in unlimited technological hubris* – McGeorge Bundy