# The Problem with Virtual

I’ve been teaching virtual and in-person classes for a few years, I didn’t go into my last job thinking of myself as a “trainer”, but the opportunity presented itself and I found myself enjoying the experience. At that job I was a developer playing non-development roles, I learned a lot, at times I took on too much responsibility, but by the end of the run learned a few things about what I’m good at and what I’m not good at.

At the previous job, the initial training effort was content focused, and it was an emergency. A class had been sold, the original authors (two capable developers who I still talk to today) delivered an awful Keynote deck that didn’t meet our quality standards, and as the guy writing the book it made sense at the time for me to dive on it. My two day trip to California turned into a two week trip, I skipped Thanksgiving, and ended up teaching a class a day after the material was finished. For a short time I was “the trainer” and I taught a number of in-person classes for that company.

That company eventually found some other trainers to teach classes, and while I still played a role in both content and teaching. Eventually that same company started to experiment with virtual training classes that became very popular. As the global economy tanked in 2008 it was clear that companies had a limited appetite for travel expenditures and virtual classes were also an easier way to get a whole team trained quickly without having to wait until the next scheduled class. So, over the past 5-6 years I’ve probably taught on the order of 200-300 students and maybe 10-20 classes. Maybe it’s more than that, but that’s estimate feel right.

Some things I learned from the experience:

**Good trainers can be a handful.** It’s tough to find developers who can communicate, and it is even more difficult to find trainers who are experts in the particular topic you are teaching. One common trait is that many trainers are often over-confident to a fault. This was evident in a few interactions over the past few years, and it was only manageable because training coordinators and sales people knew how to manage egos.

**If you enjoy training and want to do it every day you might not be the right person to teach the class.** Training can be a lucrative occupation. If you taught a full training class every day, you’ll make good money, but if you taught a training class every day you’d also have no time to use and understand the stuff you are trying to teach. This is sort of how I feel about the No Fluff Just Stuff circuit. Some of these people are really good at what they do, they all seem capable, they all seem charismatic, but more than a few of them give off a poseur vibe.

If you are doing your job right as an instructor it will be exhausting. Maybe not as true for virtual classes, but if you are invested in teaching a class and if you are going after the students to trying to keep them engaged. This class should be exhausting.

**Every trainer wants to tell you what’s wrong with the class.** There are few exceptions to this rule, and, admittedly, the class I was teaching was thrown around like an expensive football for many years. First it was the Coloradans sending us a crappy Keynote, then it was my emergency version in Keynote, then that company paid someone from New Zealand to redo the slides, our most successful trainer then decided to redo the slides and add Zen imagery, and lastly we moved the training to Asciidoc.

Along the way we had maybe ten people giving feedback about what changes needed to be made to the training. One of the reasons for this is because we didn’t do a good job agreeing on objectives and an outline up front. We also maybe didn’t do a good job structuring the effort. While there was a training coordinator and while training at that company generated revenue, the management always made every feel bad for wanting to spend any money on the effort. Don’t get me started.

Many students want to tell you what’s wrong with the class. Without exception, I’d get to the end of an in-person class only to have a couple of students stick around to tell me that they had a few, better ideas for how to teach the class.

**Many virtual students are not paying close attention.** So, don’t put people on the spot.

Lab manuals are always broken. To the point where I never had faith that the labs were actually going to work.

Your content will never be done. If you think that content is a one-time investment for a training class, think again. If you teach a class, plan on updating it at least once a year.

Typos affect the credibility of the trainer. When you are standing in front of a class of 20 students and there’s something wrong with a slide. Or, when you hit on a slide that has a typo it looks very sloppy.

Material was rarely targeted at the right audience.

**You never teach the same audience twice.** Here are two examples:

1. I had a class in California at a eCommerce company that consisted of me standing in-front of 10 students for two days answering the more complex questions you could imagine. Not one slide was delivered. It was a good class, but there was no outline for that one.
2. I had another class a few months later at a large government agency during which, over the course of two days, I only had two questions. One of them was “How do I set a windows environment variable?”

and there are a few problems with applying an in-person training approach to virtual training. The most obvious problem is that you can’t interact with students the same way.

When you teach an in-person training class, you always want to leave room for questions. In my experience, the most rewarding classes for both the student and the instructor are the classes that involve a lot of questions. When students start asking questions this means that the instructor knows they are paying attention, but it also makes the teaching experience more interesting.

In an in-person class you can see how students are responding to the material, and this gives you an opportunity to engage people directly. In a virtual environment you are talking to your laptop for a few hours. If you are lucky you’ll have a class that isn’t shy about interrupting you and you can start to elicit questions from them that help fill out the class.

If you present a slide on configuring a dependency, someone can interrupt you and ask a specific question.