

TCCI

Training Practice Transformation

Introduction – starting with your desired business outcome

TCCI's people have a breadth of experience both creating new training practices as well as transforming existing practices to meet today's needs. While each engagement is unique in some ways, they all follow these basic steps:

We work with you to identify the specific problem(s) are you trying to solve. Here are examples:

- You want to increase early customer adoption by releasing training at product release so customers can use new features immediately on release. This can bring more revenue if the feature has add-on licensing.
- You want to create in-product training for your SaaS product to slow customer defections caused by their not being able to use your new or changed features.
- You want to create e-learning to train a large base of internal employees or customers in a scalable way, but need to have the training be effective immediately and in the medium-to-long term (level 4 in the Kirkpatrick Model), instead of boring the learner to tears and having them tune out, limiting learning, effectiveness, and ultimately profits.
- You want increased training-customer satisfaction through courses that have better design, content, and development.
- You want to train your software engineers to be Agile (instead of Fragile), test-driven-development (TDD), or some new-to-them technology, increasing your engineering velocity, company agility, and profits.

Finding solutions for the problems

In our experience the solutions to these problems range from transforming small parts of a training practice to transforming the entire practice. It may involve training your designers / developers / trainers, creating written and measurable policies, standards and procedures. It may involve developing automation of lab environments to enabling performance-based e-learning and/or consistent learner experiences for instructor-led classes. We break down the problem to find the solution.

Executing on problem solutions

We have helped training organizations implement many different solutions to solve their problems. For example:

- We have led training organizations to adopt Agile methodologies for design and development of content, and delivery through e-learning, including integration with engineering organizations, allowing them to release content as product is released, including releasing in-product training for SaaS products in parallel with feature release / modification.
- We have helped training organizations start the transformation from having a pay-per-course model to having a training-as-a-service model where customers can consume as much training as they need while allowing the organization to deliver that training in a scalable manner that doesn't break their bank. This has included creating effective e-Learning whose delivery scales automatically.
- We have helped organizations develop automated lab-deployment and destruction to enable both e-Learning and instructor-led learning where historically instructors (or students) spent time setting up their own lab environments. This has cut staff time and increased training consistency and effectiveness. We've also helped create infrastructure that automatically builds / updates lab environments for each class based on the specific needs of the class.
- We have helped training organizations operating at maturity level 1 of [the Competency Maturity Model Integration \(CMMI\)](#) where things were chaotic and helped them develop policies, standards, and procedures that brought them all the way to level 5, with consistent quality, satisfied learners, and increasing velocity through continuous improvements. We have integrated the CMMI process with Agile design, development, and delivery processes.
- We have learned new technologies and methodologies needed by large engineering organizations and helped them design, develop and deliver training that enabled their engineers to dramatically increase their engineering quality, the speed of their development (by spending less time fixing bugs) and ultimately increasing profits by beating their competitors to market on new features. A common scenario has been to help engineering organizations become effective at Agile methodologies where previous attempts have failed.
- We have helped transform novice curriculum designers, developers, and trainers into highly-effective professionals. We follow the practices of [Harold Stolovich](#) in each of these disciplines of a training practice. We further his practices by integrating them with Agile methodologies and also integrating them with Agile software engineering practices, enabling parallel releases of features and content and highly-effective instructor-led training delivery.

- Because we have a deep software engineering background as well as training backgrounds, we have been able to merge training and engineering practices that allow training groups to release courses in parallel with engineering releases.

Measure, refine, iterate

- Just as with CMMI level 5, we help organizations establish methods for measuring whether the solutions we've helped them create are solving their problems to a degree that meets or exceeds requirements. When there is a miss, we help refine the solution to increase effectiveness. Where expectations are met, we help them determine if they want to refine their expectations to further improve their practice and profits, or whether they want to move on to refining other parts of their practice.

Training, training, training

- Throughout our engagement, we train our customers to how to do each of the things we help them do. For those that want to learn how to keep transforming on their own, we teach them how to do it in addition to initially doing the transformation with them.



The Computer Classroom, Inc.
629 Rinpoche Lane
Oregon WI 53575

608-835-8173
info@compclass.com