

**OLIVIER DE
WECK:**

One of the key ways of learning is to look at examples, past examples, of both successful and failed projects or designs. What we find in systems engineering is that often the problems begin at the requirements stage, where either a requirement is missing, or there's redundancy, or in some cases requirements are written that are overly ambitious and utopian given the time frame and the budget available. So what you have is always a trade off between how ambitious you are technically, the schedule, and the budget that's available, and how much risk you are willing to take.

And so the students are pushing the envelope in these different dimensions and learning what's feasible, what's not feasible, and what can we learn from past failures. We can learn from general examples of projects that had difficulties, either in aerospace, in the automotive industry, in consumer products. In hindsight, one is always smarter. But it is worth to look at these.

Specifically with the CanSat competition, since the competition has a rich history, and a very active set of sponsors, and quite a good website as well, there's a lot of opportunities to learn from mistakes that other teams have made in the past. For example, designing a glider where there's not sufficient tolerances between the vehicle itself and the container, and basically the deployment doesn't go well. Another example is designing a glider that does not fly properly in a circular pattern, and flies too well actually, and flies off into the woods never to be seen again.

There's, of course, a lot of lessons learned from establishing a stable communications link, from having worked out procedures. How do you deploy your system? How do you test it? How do you communicate with it? How do you collect it? How do you post-process the data? In each of those critical areas, we see that learning from the past is critical to avoid mistakes that could be avoided with these learnings.