

Akin's Laws of Spacecraft Design*

1. Engineering is done with numbers. Analysis without numbers is only an opinion.
2. To design a spacecraft right takes an infinite amount of effort. This is why it's a good idea to design them to operate when some things are wrong.
3. Design is an iterative process. The necessary number of iterations is one more than the number you have currently done. This is true at any point in time.
4. Your best design efforts will inevitably wind up being useless in the final design. Learn to live with the disappointment.
5. (Miller's Law) Three points determine a curve.
6. (Mar's Law) Everything is linear if plotted log-log with a fat magic marker.
7. At the start of any design effort, the person who most wants to be team leader is least likely to be capable of it.
8. In nature, the optimum is almost always in the middle somewhere. Distrust assertions that the optimum is at an extreme point.
9. Not having all the information you need is never a satisfactory excuse for not starting the analysis.
10. When in doubt, estimate. In an emergency, guess. But be sure to go back and clean up the mess when the real numbers come along.
11. Sometimes, the fastest way to get to the end is to throw everything out and start over.
12. There is never a single right solution. There are always multiple wrong ones, though.
13. Design is based on requirements. There's no justification for designing something one bit "better" than the requirements dictate.
14. (Edison's Law) "Better" is the enemy of "good".
15. (Shea's Law) The ability to improve a design occurs primarily at the interfaces. This is also the prime location for screwing it up.
16. The previous people who did a similar analysis did not have a direct pipeline to the wisdom of the ages. There is therefore no reason to believe their analysis over yours. There is especially no reason to present their analysis as yours.
17. The fact that an analysis appears in print has no relationship to the likelihood of its being correct.
18. Past experience is excellent for providing a reality check. Too much reality can doom an otherwise worthwhile design, though.
19. The odds are greatly against you being immensely smarter than everyone else in the field. If your analysis says your terminal velocity is twice the speed of light, you may have invented warp drive, but the chances are a lot better that you've screwed up.

20. A bad design with a good presentation is doomed eventually. A good design with a bad presentation is doomed immediately.
21. (Larrabee's Law) Half of everything you hear in a classroom is crap. Education is figuring out which half is which.
22. When in doubt, document. (Documentation requirements will reach a maximum shortly after the termination of a program.)
23. The schedule you develop will seem like a complete work of fiction up until the time your customer fires you for not meeting it.
24. It's called a "Work Breakdown Structure" because the Work remaining will grow until you have a Breakdown, unless you enforce some Structure on it.
25. (Bowden's Law) Following a testing failure, it's always possible to refine the analysis to show that you really had negative margins all along.
26. (Montemerlo's Law) Don't do nuthin' dumb.
27. (Varsi's Law) Schedules only move in one direction.
28. (Ranger's Law) There ain't no such thing as a free launch.
29. (von Tiesenhausen's Law of Program Management) To get an accurate estimate of final program requirements, multiply the initial time estimates by pi, and slide the decimal point on the cost estimates one place to the right.
30. (von Tiesenhausen's Law of Engineering Design) If you want to have a maximum effect on the design of a new engineering system, learn to draw. Engineers always wind up designing the vehicle to look like the initial artist's concept.
31. (Mo's Law of Evolutionary Development) You can't get to the moon by climbing successively taller trees.
32. (Atkin's Law of Demonstrations) When the hardware is working perfectly, the really important visitors don't show up.
33. Space is a completely unforgiving environment. If you screw up the engineering, somebody dies (and there's no partial credit because *most* of the analysis was right...)

*I've been involved in spacecraft and space systems design and development for my entire career, including teaching the senior-level capstone spacecraft design course, for ten years at MIT and now at the University of Maryland for more than a decade. These are some bits of wisdom that I have gleaned during that time, some by picking up on the experience of others, but mostly by screwing up myself. I originally wrote these up and handed them out to my senior design class, as a strong hint on how best to survive my design experience. Months later, I get a phone call from a friend in California complimenting me on the Laws, which he saw on a "joke-of-the-day" listserve. Since then, I'm aware of half a dozen sites around the world that present various editions of the Laws, and even one site which has converted them to the Laws of Certified Public Accounting. (Don't ask...) Anyone is welcome to link to these, use them, post them, send me suggestions of additional laws, but I do maintain that *this* is the canonical set of Akin's Laws...