

8th class: Rittel and wicked problems

Thursday, November 15, 2007
7:10 PM

Rittel, H., and Webber, M., "Dilemmas in a General Theory of Planning", Reprint 86, University of California at Berkeley, Institute of Urban and Regional Development.

Rith, C., and Dubberly, H., "Why Horst W. J. Rittel Matters," Design Issues, Volume 23, Number 1, MIT Press, Cambridge, 2007.

Hugh needs to be in Seoul the first week in December. So we'll be switching it to Dec 11.

Rittel

We review the three basic types of problems:

- Simple: Clear goal/problem, you can map it pretty easily onto a feedback loop of method/action and measure/test
- Complex: Goal is not yet clear, but it is possible to agree on it. Agreeing transforms it into a simple problem. (This is why you do planning meetings with all sorts of stake-holders to get "buy in".)
- Wicked: You can't agree on the goals, so if you want to turn it into a complex problem you're going to have to reframe and innovate.

The key thing in practice is to know which kind of problem you're engaged in (and have a discussion with the client about it).

Is Rittel a constructivist? Yep. His basic idea of wicked problems means that there are multiple (valid) view points on these problems so that you can't take it for granted that there's a single objective goal that's right (and the others are wrong).

What are the consequences? Well, you need to get people to see the different view points and not immediately try to jump into solutions (which always presuppose goals). You have to get sympathy about the different sets of goals being expressed and maybe brainstorm about others that might exist for people not at the table.

Thus you're moving away from "design as problem solving", where there's a right answer (where your just searching through the solution space; optimizing, satisfying, and all sorts of Simonesque activities). There is a stopping condition.

There's a model of design as art, "make it look good", where the designer optimizes for himself. This is probably the earliest.

You can then move towards user-centered design (which HD values), but he's still considering this problem solving. Perhaps, says the class, this can push towards conversation, but the basic part of user-centered design involves "figuring out the user" and

Miscellaneous

Frog's philosophy is that "form follows emotion". But who's emotion? The designers.

See Alan Cooper (father of visual basic), he has a firm that does software design (and is well regarded). *About Face* is probably the most famous. In it he popularized personas. He is quite explicit that you should go observe the users, define the problem, go do the design and keep the user's goals in mind, but don't keep them around because they are confusing and confused. They can't generalize, so ban them.

Haggai points out that you can do a lot of stuff with that view point.

Steve points out that at Apple you hire the best designers and trust that they will tell you what the users want, not needing to have the user nearby.

Google's design tends to be a conversation but only amongst engineers.

HD suggests that it's a huge management problem to have the right people on the problem--the right amount of requisite variety. Victoria points out that it's hard to not have a general policy about who meets to solve problems.

"As You May Think" was written by Bush. He was Roosevelt's national science advisor. Right after the war, he essentially envisioned the Internet in an article (taking documents and linking related parts of them, he also envisioned recording paths through this). Hypertext starts here.

"All models are wrong, some are more wrong than others," says HD.

History

Operations research (OR) and cybernetics were partly responsible for technology euphoria after the war and into the 1960s.

In 1938, the winds of war were gathering in Europe. The British knew they were going to go to war; they had some smart scientists who had invented radar.

just adding criteria to the problem space. I wonder about this, since in practice you always have debates about the users and sometimes with.

Finally, there is design as conversation/argument/rhetoric/deliberation. It falls out of this that design is political. Whereas in the problem-solving model the "therefore" is that "therefore design is rational and objective". (The "therefore" for design as art is "therefore design is subjective".)

HD feels that it is very valuable to be able to frame design in this way as you talk with your manager, your employees, your VPs, etc. You need to be able to ascertain where the people who work for you are.

HD says there's movement in Rittel's direction but that mainly design is a quagmire not moving forward. NYU is still in design as art. IIT is in the design methods/problem-solving area. There's NO school where design as conversation is taught, though people do have to figure it out.

Are we clear on why everyone's an expert (or no one is)? Rittel has the phrase "symmetry of ignorance" for this. The idea here is that you define problems in the areas you're expert in. The business and the users have insights, too. If you have a designer as an expert, then what expertise is s/he bringing? They are being experts in problem solving. The problem is understanding the current situation. But you have to agree on what the problem is.

Gus wants to say that each mode does have situations where it works best. HD says that's fair.

Gus's project on the iPhone

Is the iPhone innovative? Gus doesn't think it is really that remarkable. But he's interested in the malaise of the handheld industry over the last decade. (By traditional business measures, phone companies have been very successful, but for consumers, the companies have walled gardens that stifle innovation.)

Gus's basic idea is that the original Handspring Treo is the last thing that was innovative and that was a long time.

Gus shows a model of organizational processes for Apple where there's a very controlled system with Steve Jobs giving a lot of guidance from on high. Hugh adds a little nuance here: people who work with Steve tell him that the amount of guidedness varies depending on whether it's a consumer product or technical one. He takes a more active role in consumer areas. Gus points out that OSX falls in between.

They had set up radar along the coast of SE England and at the research station they had engineers keeping it working. The guy running the thing said let's see how it does. They sent the blue squadron and saw that they came back. But the operations didn't work (no one really gave the feedback). So the scientists worked to figure out how to get the planes to go meet those. OR is basically complex math used to solve practical problems (if you've got convoys moving to the UK and you have a limited number of destroyers, how do you optimize the destroyers' path). This worked well.

After the war, we start building missiles, rockets, spaceships, nuclear submarines, etc. HD maintains that Ulm is a central place where this comes together after the war. Rittel and others were soaking in these ideas.

As they tried to apply these models to social problems, they did terrible things like building highways through New Haven. They got chewed out immediately. Venturi wrote about complexity and post-modernism. OR ideas started to fade. Alexander and Jones repudiated the stuff. Archer faded away. Rittel tried to rehabilitate it with second-generation design methods, which HD says parallels nicely what's going on in cybernetics at the time.

Steve's debugging

He spent the weekend watching novice programmers. There's actually a call for papers for next year because people don't know about how people actually write and debug (though there's lots of info about expert code writing/debugging). Steve's also been a tutor, so he probably has as much domain expertise as anyone.

HD feels like he's in a rathole about how you can do research on this. What is the mental model and how can you know it? Giving them a piece of paper to draw their mental model didn't work (this given to expert Java programs).

HD: You could use a think aloud protocol, too.

This makes me think about Latour and his claim that it's by perturbing people's categories that they reveal them.

Matt's tag cloud

It's been done! Try it out:
<http://www.tagcrowd.com/>

Matt's project will look at mental models that will further refine these visualization--perhaps by showing excerpts (so you can see the connections of words to the words in the article); you can also drill down to words closer to instances.

Gus also models Palm OS, Palm Corporate, and Carriers. He shows how Palm stagnates (someone points out that Palm was truly a mess).

HD: You're getting at a problem of "design as a conversation" when it comes to deciding about vision. (This is the question HD would ask Rittel if he came back from the dead for half an hour.)

Gus also shows a model of end-user feedback and where different companies break down.

Note for Gus: Maybe change the word "create more variety" to "novelty" per Sean's suggestion (for the model of carrier influence).

Gus proposes that carriers were scared of results of dot com explosion where they had a bunch of capacity that they wanted to do.

HD counter-proposes that these carriers are children of telcos, which are just basically non-innovative, wanting to be monopolistic and minimize risk (operations mentality instead of developer mentality is Gus's idea).

HD says it's odd to have carriers above and below. An entity should only be in one place. Minus ten points.

HD likes the demonstration of how feedback can be screwed up.

HD asks about link of Apple to AT&T. No one likes their carrier. The field is highly regulated also. Some talk about VPNO and some other access points.