Wicked Problems and Fragmentation[1]

by Jeff Conklin, Ph.D.

Some problems are so complex that you have to be highly intelligent and well informed just to be undecided about them.
--Laurence J. Peter

The first step in improving collaboration is to understand the barriers to collaboration. This paper describes the chief barrier to collaborative success—fragmentation—and shows how fragmentation is caused by wicked problems and social complexity. Fragmentation, for example, is when the stakeholders in a project are all convinced that their version of the problem is correct. The way to combat fragmentation is through shared understanding and shared commitment, which is inherently a social process.

Organizational Pain

There is a subtle but pervasive kind of pain in our organizations. It is characterized by such frequently heard complaints as, "How am I supposed to get my work done with all of these meetings?" and "We always have time to do it over again, but never time to do it right." It is a sense of futility of expecting things to be one way and repeatedly banging into a different reality. It is the dull

[1] **This paper is Chapter 2 in *Dialog Mapping: Making Sense of Project Fragmentation* by Jeff Conklin, forthcoming.**

jeff@cognexus.org
CogNexus Institute
http://www.cognexus.org

ache of *deja vu* when you are handed an impossible deadline or a vague assignment. It is the frustration of calling a meeting to make a decision and watching the meeting unravel into a battle between rival departments, or get lost in a thicket of confusion over the meaning of a technical term. It is the frustration of finally achieving a hard-won decision and then having it fall apart or get "pocket vetoed" because there wasn't really buy-in.

I was working late one evening when the janitor came in to vacuum the office. I noticed that he was going back and forth over the same areas without appearing to get the lint up off the carpet. I smiled and shouted to him (the vacuum cleaner was a loud one) "It must be frustrating to have to use that vacuum cleaner." He looked at me with a sad smile and said "Not as frustrating as being told to go back and do it over!" It is *that* kind of pain, and it goes all the way up to the executive suite.

Part of the pain is a misunderstanding of the nature of the problems at hand. More precisely, the pain is caused by working on a new class of problems—*wicked problems*—with thinking, tools, and methods that are useful only for simpler ("tame") problems. Most projects today have a significant wicked component. Wicked problems are so commonplace that the chaos and futility that usually attend them are accepted as inevitable. Failing to recognize the "wicked dynamics" in problems, we persist in applying inappropriate methods and tools to them.

The challenge of wicked problems is exacerbated by *social complexity*—the number and diversity of stakeholders who are part of the problem solving process. Social complexity means that the environment of a project team is populated by individuals, other project teams, and other organizations which have the power to undermine the project if their stake is not considered—if they are not at least included in the thinking and decision making process.

My janitor friend had an advantage over the rest of us in the organization because he could clearly see that his vacuum cleaner was not actually picking up the dirt. When we are working on wicked problems in a socially complex environment, it is much harder to notice that our tools are simply not "picking up the dirt."

So what is a wicked problem? Before answering that question, we first step into the laboratory to take a closer look at the process of attempting to solve an unfamiliar problem.

The MCC Elevator Study

A study in the 1980's at the Microelectronics and Computer Technology Corporation (MCC) looked into how people solve problems. The study focused on design, but the results apply to virtually any other kind of problem solving as well, including planning and analysis.

A number of designers participated in an experiment in which the exercise was to design an elevator control system for an office building. All of the participants in the study were experienced and expert integrated circuit designers, but they had never worked on elevator systems before. Indeed, their only experience with elevator systems came from riding in elevators. Each participant was asked to think out loud while they worked on the problem. The sessions were videotaped and analyzed in great detail[1].

The analysis showed, not surprisingly, that these designers worked simultaneously on *understanding the problem* and *formulating a solution*. They exhibited two ways of trying to *understand the problem*:

- Efforts to understand the requirements for the system (from a one page problem statement they were given at the beginning of the session)
- Mental simulations (e.g., "Let's see, I'm on the second floor and the elevator is on the third floor and I push the 'Up' button. That's going to create this situation....").

On the solution side, their activities were classified into *high*, *medium*, and *low* levels of design, with high-level design being general ideas, and low being details at the implementation level. These levels are analogous to an architect's sketch, working drawings, and a detailed blueprint and materials list for a house.

Traditional thinking, cognitive studies, and the prevailing design methods all predicted that the best way to work on a problem like this was to follow an orderly and linear process, working from the problem to the solution. This logic is familiar to all of us. You begin by understanding the problem. This often includes gathering and analyzing "requirements" from customers or users. Once you have the problem specified and the requirements analyzed, you are ready to formulate a solution, and eventually to implement that solution. This is illustrated by the "waterfall" line in Figure 1.

This is the pattern of thinking that everyone attempts to follow when they are faced with a problem, and it is widely understood that the more complex the problem is, the more important it is to follow this orderly flow. If you work in a large organization, you will recognize this linear pattern as being enshrined in policy manuals, text books, internal standards for project management, and even the most advanced tools and methods being used and taught in the organization. In the software industry it is known as the "Waterfall model," because it suggests the image of a waterfall as the project "flows" down the steps towards completion.

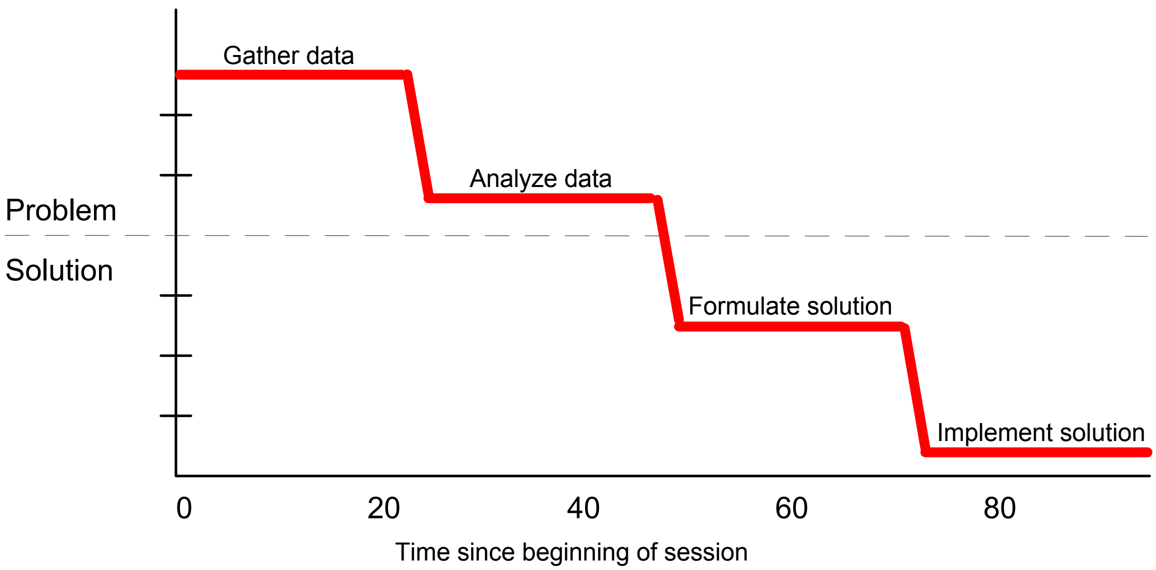

Figure 1: Traditional wisdom for solving complex problems: the "waterfall"

However, the subjects in the elevator experiment did not follow a waterfall. They would start by trying to understand the problem, but they would immediately jump into formulating potential solutions. Then they would jump back up to refining their understanding of the problem. Rather than being orderly and linear, the line plotting the course of their thinking looks more like a seismograph for a major earthquake, as illustrated in Figure 2. We will refer to this jagged-line pattern as both *chaotic*, for obvious reasons, and *opportunity-driven*, because in each moment the designers are seeking the best opportunity for progress toward a solution.

These designers are not being irrational. They are not poorly trained or inexperienced. Their thought process was something like: "Let's see, idle elevators should return to the first floor, but then, you only need one elevator on the first floor, so the others could move to an even distribution among the floors. But the elevators need to be vacuumed regularly. I suppose we could add a switch that brought idle elevators down to the first floor. But then what happens in an emergency?" In other words, what is driving the flow of thought is some marvelous internal drive to make the most headway possible, regardless of where the headway happens, by making *opportunity-driven* leaps in the focus of attention. It is precisely because these expert designers are being creative and because they are learning rapidly that the trace of their thinking pattern is full of unpredictable leaps.

In particular, the experiment showed that, faced with a novel problem, human beings do not simply start by gathering and analyzing data about the problem. Cognition does not naturally form a pure and abstract understanding of "the problem." The subjects in the elevator experiment jumped immediately into thinking about what kind of processors to use in the elevator controller, and how to connect them, and how to deal with unexpected situations, such as if one processor failed. These are detailed solution elements.

These experienced designers illustrated that problem understanding can only come from creating possible solutions and considering how they will work. Indeed, the problem often can best be described in terms of solution elements. A requirement in the problem statement calling for "high reliability" was quickly translated into the idea of using a network of distributed processors—a high-level solution that drove the rest of the design process.

It is also striking to see, from Figure 2, that problem understanding continues to evolve until the very end of the experiment. Our experience in observing individuals and groups working on design and planning problems is that, indeed, their understanding of the problem continues to evolve—forever! Even well into the implementation of the design or plan, the understanding of the problem, the "real issue," is changing and growing.

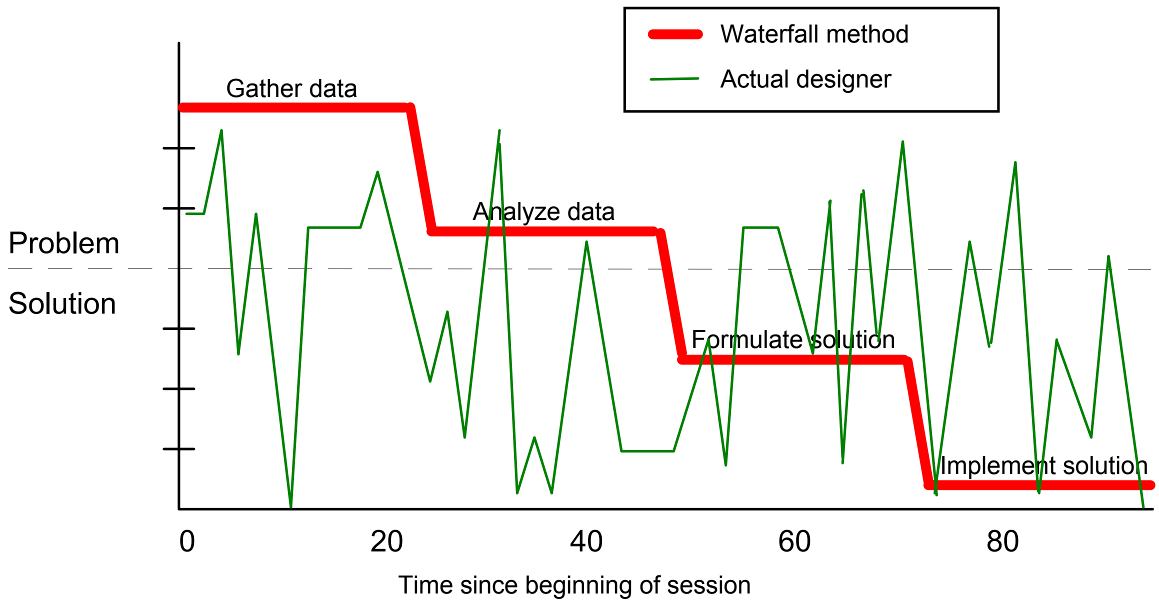

Figure 2: Pattern of cognitive activity of one designer -- the "jagged" line

The natural pattern of problem solving behavior may appear chaotic on the surface, but it is the chaos of an earthquake or the breaking of an ocean wave—it reveals deeper forces and flows which have their own order and pattern. The non-linear pattern of activity that expert designers go through gives us fresh insight into what is happening when we are working on a complex and novel problem. It reveals that it is not a mark of stupidity or lack of training that we seem to "wander all over." This non-linear process is not a defect, but rather the mark of an intelligent and creative learning process.

As chaotic as this pattern of activity appears, it reflects a deeper order in the cognitive process. It shows that people formulate possible solutions and try them out in order to better understand the problem. The new insights into the problem gave them fresh ideas about the shape of the solution.

In fact, this non-linear pattern does not come as a surprise to most people. Anyone who has ever worked on a hard problem has the intuition that this jagged line process is what they are really doing. But the experiment is significant because, for the first time, we have a picture of the process that people follow when they really think about novel problems, and it is not the orderly and linear process we have been taught is proper!

From another perspective, the jagged line is a picture of *learning*. The more novel the problem, the more the problem solving process involves learning about the problem as much as designing a solution. In this sense the waterfall is a picture of *already knowing*—you already know about the problem, you know about the right process and tools to solve it, and you know what a solution will look like. However, in a novel and knowledge-driven environment, problem solving and learning are tightly intertwined, and the flow of this exploratory process is chaotic and opportunity-driven, as the jagged seismic line illustrates. Thus, the MCC elevator experiment gives us a graphical insight into the process of working on "*wicked problems*," because a wicked problem is, among other things, always novel.

Wicked Problems Defined

The man who coined the term "wicked problem," Horst Rittel, was also the inventor of the Issue-Based Information System (IBIS) structure upon which the Dialog Mapping technique [described in the book] is based. Thus this work owes everything to Rittel's genius about the nature of the design process[2]. Rittel and his colleagues perceived the limitations of the linear "systems approach" of design and planning over 30 years ago, and their research provides a foundation for what Rittel termed a "second generation" of systems analysis methodology. Rittel invented IBIS because, as an urban planner and designer, he found traditional planning methods inadequate for the ill-structured problems he encountered in city planning.

Rittel's genius shines especially bright when we consider his solution for wicked problems: IBIS, a structure for rational *dialog* among a set of diverse stakeholders. This is a perspective that puts human relationships and social interactions at the center, a perspective that is only now coming into vogue as a key insight of post-modern thought. This paper is really just an extension of the work that Rittel started.

As Rittel defined them[3], wicked problems are distinguished by the following criteria:

1. *You don't understand the problem until you have developed a solution.*

   Indeed, there is no definitive statement of "The Problem." The problem is ill-structured, an evolving set of interlocking issues and constraints. Rittel said, "One cannot understand the problem with knowing about its context; one cannot meaningfully search for information without the orientation of a solution concept; one cannot first understand, then solve." Moreover, what "The Problem" is depends on who you ask—different stakeholders have different views about what the problem is and what constitutes an acceptable solution.[4]

2. *Wicked problems have no stopping rule.*

   Since there is no definitive "The Problem," there is also no definitive "The Solution." The problem solving process ends when you run out of resources, such as time, money, or energy, not when some optimal or "final and correct" solution emerges. Herb Simon, Nobel laureate in economics, called this "satisficing" -- stopping when you have a solution that is "good enough" (Simon 1969).

3. *Solutions to wicked problems are not right or wrong, simply better, worse, good enough, or not good enough.*

   With wicked problems, the determination of solution quality is not objective and cannot be derived from following a formula. Solutions are assessed in a social context in which "many parties are equally equipped, interested, and/or entitled to judge [them]," and these judgments are likely to vary widely and depend on the stakeholders' independent values and goals.

4. *Every wicked problem is essentially unique and novel.*

   There are so many factors and conditions, all embedded in a dynamic social context, that no two wicked problems are alike, and the solutions to them will always be custom designed and fitted. Rittel: "The condition in a city constructing a subway may look similar to the conditions in San Francisco, say, but differences in commuter habits or residential patterns may far outweigh similarities in subway layout, downtown layout, and the rest." Over time one acquires wisdom and experience about the approach to wicked problems, but one is always a beginner in the specifics of a new wicked problem.

5. *Every solution to a wicked problem is a "one-shot operation."*

   Every attempt has consequences. As Rittel says, "One cannot build a freeway to see how it works." This is the "Catch-22" about wicked problems: you can't learn about the problem without trying solutions, but every solution you try is expensive and has lasting unintended consequences which are likely to spawn new wicked problems.

6. *Wicked problems have no given alternative solutions.*

   There may be no solutions, or there may be a host of potential solutions that are devised, and another host that are never even thought of. Thus, it is a matter of *creativity* to devise potential solutions, and a matter of *judgment* to determine which are valid and which should be pursued and implemented.

Here are some examples of wicked problems:

- How to deal with crime and violence in our schools?
- Whether to route the highway through our city or around it?
- What should our mission statement be?
- What features should be in our new product?

Example: A New Car Design

Let's consider a potentially wicked problem in the design of a new car. Let's imagine a project team that has formed around a new assignment: the Marketing department is asking for a design that emphasizes side-impact safety—they want to promote a new "safe car" to compete with Volvo. That is the problem to be solved; that is the work of the project. There is a deadline and a budget and a senior executive that the project reports to.

Now consider the criteria for a wicked problem again:

You don't understand the problem until you have developed a solution.

One approach to making a safer car would be to add structural support in the doors to make the car safer from side impact. It turns out that the additional door structure doubles the cost of the door, makes the doors heavier and harder to open and close, changes the fuel mileage and ride, and requires an adjustment to the suspension and braking systems. Making the doors stronger leads into other design problems, but also bounces back into marketing problems such as, "What should the price be?" "How much do people really care about side impact survivability?" "What do customers really want in a car?" All of these problems interact with each other. And at the senior executive level, the real question is, "Should we produce this new car?"

Wicked problems have no stopping rule.

When does the car become "safe?" There is no natural stopping point in working out the tradeoffs among safety, performance, appearance, and cost. At some point, the design team will be forced to make a decision. If it were not for project deadlines, the team would swirl indefinitely in "analysis paralysis."

Every wicked problem is essentially unique and novel.

Even if the project team has several successful car designs under its belt, the "safe door" problem is essentially unique and novel, because of the configuration of issues and stakeholders. First, a recent study by a consumer safety organization suggests that side impact injuries would be reduced by side air bags, which are not a part of the design. Second, a side-impact injury law suit has been filed against the company—if the new design is announced now, it may look like an acknowledgement of prior unsafe designs. Moreover, federal legislation is emerging that may put legal constraints on the strength of the doors. The design of safer doors is not merely a technical problem: it is a political and PR problem as well.

Every solution to a wicked problem is a "one-shot operation."

The creation of a safer car is a one-shot operation. When the new safer car finally reaches the market, it may be a flop, or it may change the safety standards for the whole industry. The design team can build prototypes of the car and test them, but there is no way to anticipate the unintended consequences of producing and selling the new vehicle.

Wicked problems have no given alternative solutions.

The safe door problem does not have a few discrete possible solutions from which to choose. There is an immense space of options in terms of structural reinforcement, materials, window design, hinge placement, and how the door latches and opens. The design team cannot select from a few options—it must collectively exercise creativity and judgment about an elegant resolution of all of the design priorities.

The design of a new "safe car" is an example of a wicked problem. It cannot be solved by engineers alone, nor is there any way of determining that any given solution is "correct" or even optimal. It all depends on where you stand.

Coping with Wicked Problems

Not all problems are wicked. In contrast, a *"tame problem"* is one for which the traditional linear process is sufficient to produce a workable solution in an acceptable time frame. A tame problem:

1. Has a well-defined and stable problem statement
2. Has a definite stopping point, i.e. when the solution is reached
3. Has a solution which can be objectively evaluated as right or wrong

4. Belongs to a class of similar problems which are all solved in the same similar way
5. Has solutions which can be easily tried and abandoned
6. Comes with a limited set of alternative solutions

Finding the square root of 58 is a tame problem, as is finding the shortest route from A to B on a map. Fixing a television set, raising $1000, and selecting a new doctor when you move to a new city are all tame, if messy, problems. Tame does not mean simple—a tame problem can be very technically complex. Even putting a man on the moon was a fundamentally tame problem, although it may have had wicked sub-problems. If the problem statement doesn't change over time and there is no disagreement over it, and if it is clear when a possible solution does or does not solve the problem, then the problem is tame.

It can be difficult to tell a tame problem from a wicked problem. Like the safe car design example, many problems appear tame on the surface, but are indeed wicked once you get into them. Issues and problems in the real world occur on a spectrum from tame to wicked, and there is a natural human desire to have problems be tame and to avoid the wicked ones.

As with anything that has been held in denial, the first step in dealing with a wicked problem is to recognize it. There is a tendency to treat all problems as tame, perhaps because tame problems are easier to solve, reinforced by the lack of understanding about wicked problem dynamics and the tools and approach they require.

The command and control paradigm of management reinforces blindness about the true nature of the problem. Inherent in this paradigm is the idea that a person in charge gives the solution (the right solution, the only solution) to other people, who are in charge of implementing it. To function in such a hierarchy often means to collude in systematic denial of the complex and ill-structured dynamics of wicked problems, a phenomenon dubbed "skilled incompetence" by Chris Argyris (Argyris and Schön, 1996).

As a result, there are two common organizational coping mechanisms that are routinely applied to wicked problems: *studying* the problem and *taming* it.

While studying a problem is natural and important, it is an approach that will fail if the problem is wicked. Study amounts to procrastination, because little can be learned about a wicked problem by mere study. As we saw, learning about wicked problems requires making decisions, doing experiments, launching pilot programs, testing prototypes, and so on. Study alone leads to more study, and results in the condition known as "analysis paralysis," a Catch-22 in which we can't take action until we have more information, but we can't get more information until someone takes action.

Taming a wicked problem is a very natural and common way of coping with it. Instead of dealing with the full wickedness of the problem, one simplifies it in various ways to make it more manageable—to make it solvable! There are (at least) six ways to tame wicked problems, corresponding to the six criteria for wickedness:

1. *Lock down the problem definition.*

   Develop a description of a related problem or a sub-problem that you *can* solve, and declare that to be the problem. Resist all efforts to expand or modify the problem definition. For example, if the problem is how to reduce violence in schools, you could focus on the much more tractable problem of how to install metal detectors in all school entrances. As another example, in the software field, one learns to "freeze the requirements" as a way to lock down the problem[5].

2. *Declare the problem solved.*

   Since a wicked problem has no definitive solution, the whole point of attempting to tame it is so that a solution can be reached. Usually this step requires locking the problem down (see point 1), though it is possible to simply declare the problem "solved" without clarity about what the problem was. Such declarations, however, generally require considerable authority, such as in an autocratic organization or a dictatorship, to appear successful.

3. *Specify objective parameters by which to measure the solution's success.*

   This is the measurement approach. For example, to find out if we have solved the problem of school violence, we might count the number of deaths and injuries on school property—if this measure drops to zero, then we have solved the problem. This taming approach amounts to locking the problem down (point 1), however, because what is measured becomes, officially and by definition, the problem. Whatever is not measured is then free to absorb the real problem. With intense enough focus, we might reduce the number of violent incidents on the school grounds to zero: problem solved! But we might overlook new problems that have resulted, such as a sharp rise in violent incidents just off of the school grounds.

4. *Cast the problem as "just like" a previous problem that has been solved.*

   Ignore or filter out evidence that complicates the picture. Refer to the previous solution of the related problem: "It's just like that problem. Just do the same thing again."

jeff@cognexus.org
CogNexus Institute
http://www.cognexus.org

5. *Give up on trying to find a good solution to the problem.*

   Just follow orders, do your job, and try not to get in trouble. Maybe the organization will fix the serious problems in the current solution in a revised version or release next year.

6. *Pretend that there are just a few possible solutions, and focus on selecting from among these options.*

Different people prefer different coping mechanisms—some would rather study the problem until they really understand it, others, impatient with sitting around, would rather tame the problem to something manageable and jump into action.

However, attempting to tame a wicked problem, while appealing in the short run, fails in the long run. The wicked problem simply reasserts itself, perhaps in a different guise, as if nothing had been done. Or the tame solution may have exacerbated the problem.

Social Complexity

One factor that increases the wickedness of a problem is the *number* and *diversity* of stakeholders involved. A stakeholder, as the term implies, is someone who holds a stake in the outcome. Stakeholders are players in the social network around a project, and their input and participation is important to the project's overall success. Stakeholders often have the power to stop, undermine or even sabotage a project if it threatens them or their designated organizational role.

The MCC elevator study can help us visualize the impact of multiple stakeholders on a wicked problem. Suppose we add a second engineer, represented by the medium bold line in Figure 3, to help solve the elevator design problem.

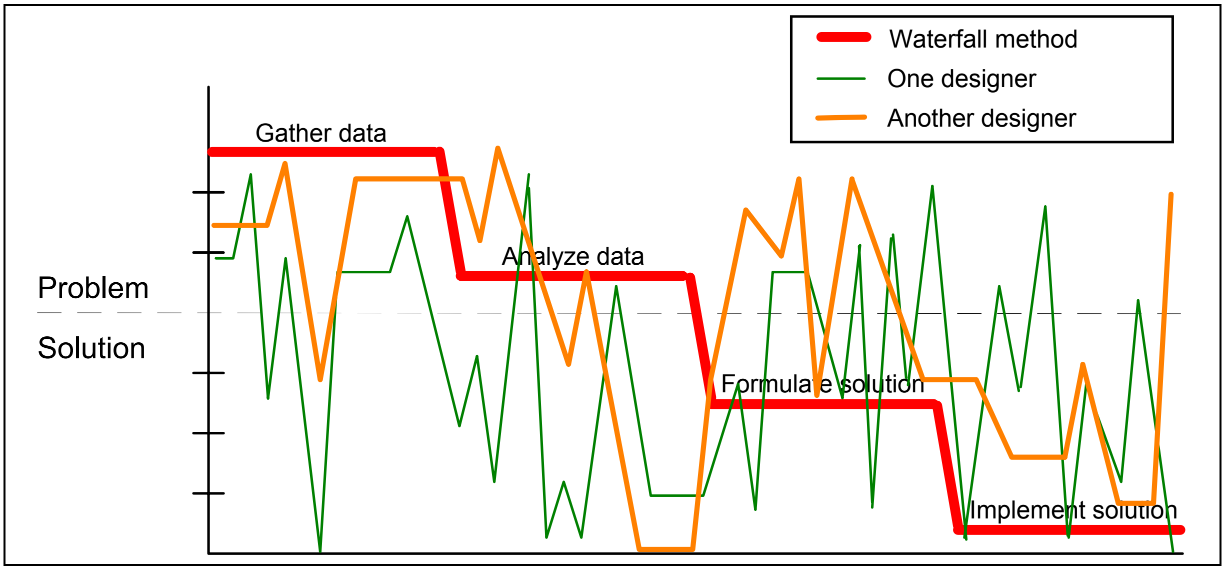

Figure 3: A wicked project with a second designer working on the problem

Notice that the new designer, like the first designer, goes through a seemingly chaotic process between the problem and solution spaces, but the new designer's specific thinking process is quite different from the first designer's. Since her background and approach are quite different, the pattern of her flow of creative thinking will also differ.

Let's imagine you are the project leader. You are the one who is responsible for the project being on time, in budget, and meeting all of its requirements—you are officially in charge of keeping the project on the waterfall line.

Now let's consider two project team meetings. At meeting A in Figure 4, you are a happy project leader because everyone is in synch, focused on the same activity, analyzing the requirements just like it says in "the book." Your prospects for bringing your project in on time and in budget look good.

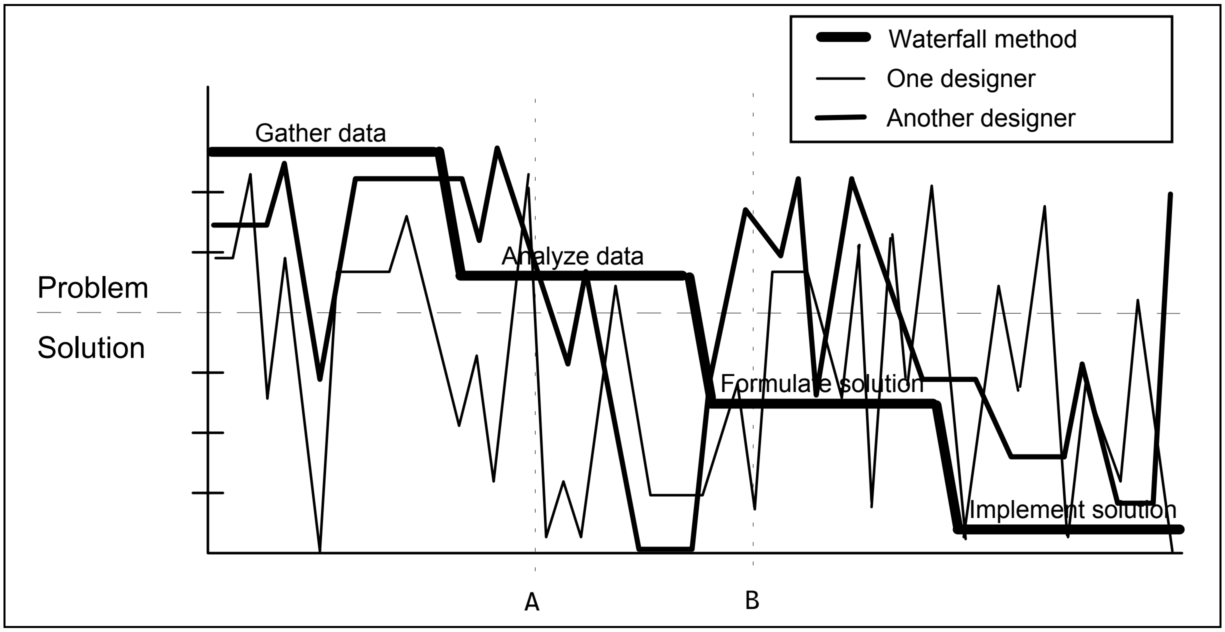

Figure 4: Two team meetings, A and B, during the project.

Some time later, at meeting B in Figure 4, the team has finished up with the data analysis and is now in the next phase, high-level design. But there are signs of trouble. One designer (represented by the thin line) looks tired but radiant. He says, "I was driving home last night and I had an idea. I stayed up all night programming, and—you won't believe this, but—I put together a program that does the whole thing. Sure, it still needs a little work, but, hey, we're practically done! Way ahead of schedule! I can't wait to show it to you!" He has made a major leap, all the way to bottom of the chart, to the final solution.

There is a long pause. Your other designer (represented by the thick seismic line) also looks tired, but not so radiant. Holding up the long-finished requirements document she says, "Sorry, we're not even close to done. I was with the client yesterday, and it turns out that there is a transaction that the system needs to handle that they never even told us about. They said it didn't have anything to do with our system. But it turns out it has *a lot* to do with our system. We've got to back to square one and start over!"

Neither of these key players is where you need them to be, according to the linear plan you created at the beginning of the project. You can feel chaos rising and control slipping away. You desperately plead with them to refocus on the high-level design, because, according to the calendar, that's where the project needs to be.

Perhaps you turn to the first designer and say something like, “That's a good idea, Henry—but we really need to finish the high-level design. Can you hang on to that code for a while?” Turning to the other designer, you beg, “Look, Sally, we already have gotten those requirements signed off. We can’t go back. We’ll just have to take care of that new transaction in the next release of the system.”

This scenario exemplifies the tension that is inherently part of working on a wicked problem. Despite the most carefully thought out plans, wicked problem dynamics tend to enhance the experience of chaos and fragmentation on a project. The above scenario is mild, however, because there are only three stakeholders involved in the project. As projects grow in size and organizations grow flatter, social complexity increases. Large projects typically have dozens of stakeholders, representing the project team, other departments, and other organizations. And not only does each one have their own “jagged line,” they are likely to have different ideas about what the real issues are, and what the criteria for success are.

Consider the safe car design team. Bob, from Marketing, has been conducting studies and focus groups that indicate a lot of interest in cars that are safer in a collision. He is concerned with how to package a new “safe car” in a way that is positive, sexy, and up-beat. Christine, from Engineering, is very concerned about making the doors too heavy, but she has worked on structural integrity in the past and is excited about new technologies that, while expensive, could make the doors both stronger and lighter. Harry, the representative from the Management Team, sees the big issue as cost—top management is pushing affordability and value as the new strategy to increase sales. Alan, from IT, has a mandate from his management to get this team to use the new CAD (Computer Aided Design) system on this project. There are team members who represent Regulatory Affairs, Finance, Graphic Design, Power Train, and Quality Assurance, as well as team members from several major suppliers, including electronics and interior materials.

Each player has their own individual experience, personality type, and style of thinking and learning. Each player adds a new jagged line to the graph. The individual diversity among these players will make collective intelligence a challenge, and will make consensus difficult to achieve.

But social complexity doesn’t stop with individual diversity – each of these players comes from a different *discipline*, with its own specialized language and culture. When Bob is among his colleagues in Marketing, they share common knowledge, a common set of concerns, and shared ways of thinking about and dealing with those concerns. However, when Bob tries to talk to Christine from Engineering, he finds that she has little knowledge of basic marketing concepts, and seems to be uninterested in them. It’s as if she were from a different country, speaking a different language. Thus, achieving shared meaning and shared context is especially difficult.

Moreover, social complexity goes beyond individual diversity and diversity among disciplines. The real corker is that these players represent different *organizations*. Each organization has its own function and charter, its own goals, and is managed by its own executive director. These organizations often have divergent goals. Marketing is trying to make its sales numbers, while Engineering is trying to win the Baldridge quality award. When the members of a project team come together to collaborate, they represent not only themselves but also their respective management chain in the hierarchy. Ideally, everyone in the organization is committed to the same thing, but operationally goals and agendas can be quite fragmented.

Thus, social complexity makes wicked problems even more wicked, raising the bar of collaborative success higher than ever.

For example, the main feature of a wicked problem is that you don't understand the problem until you have a solution. But with social complexity, "not understanding the problem" does not show up as innocent wonder about the mystery of the problem, nor does it usually occur as a thoughtful collective inquiry into the deeper nature of the problem.

Rather, "not understanding the problem" shows up as different stakeholders who are certain that *their* version of the problem is correct. In severe cases, such as many political situations, each stakeholder's position about what the problem is reflects the mission and objectives of the organization (or region) they represent. In such cases there is a fine line between collaboration and colluding with the enemy. How can you make headway on a mutually acceptable solution if the stakeholders can not agree on what the problem is?

The answer to this question—and the Holy Grail of effective collaboration—is in *creating shared understanding about the problem and shared commitment to the possible solutions*. Shared understanding does not mean we necessarily *agree* on the problem, although that is a good thing when it happens. Shared understanding means that the stakeholders understand each other's positions well enough to have intelligent dialog about the different interpretations of the problem, and to exercise collective intelligence about how to solve it.

Because of social complexity, solving a wicked problem is *fundamentally a social process*. Having a few brilliant people or

jeff@cognexus.org
CogNexus Institute
http://www.cognexus.org

the latest project management technology is no longer sufficient.

This paper offers a practical approach to the mechanics of creating shared understanding and shared commitment in a complex social network. But before we can get into the "solution," there is a little bit more to understand about the collaborative challenge posed by wicked problems and social complexity.

Simple Social Networks and the Polarity of Design

It's not necessary to have dozens of conflicting stakeholders for social complexity to present challenges. In fact, all that is needed are representatives of the two fundamental polarities of design: what is needed, and what can be built.

Virtually all creative work is a process of *design*[6]. All problems call for *designing* a solution. All projects are essentially *designing* something. Design, in both the technical and artistic sense, is the process of creating something new—e.g., a new car, a strategic plan, a software program, a clearer web site, next year's budget, a new environmental policy.

Any design problem is a problem of resolving tension between what is needed and what can be done. On the one hand, the process of design is driven by some desire or need—someone wants or needs something new. The need might be expressed by a customer, or it may be a guess about what the market wants. The need or want is expressed in the language of *what ought to be*—what should be done, what should be built, what should be written. On the other hand, the process of design is constrained by resources—*what can be done,* given the available resources such as time and money and the constraints imposed by the environment and the laws of science?

Every need has a price tag—the process of design is about devising solutions that are feasible and cost effective. Going back to the safe car design, the need might be quite specific, e.g., the car must protect the occupants from harm if it is struck from the side by another vehicle of similar weight traveling 30 miles an hour. It may turn out that such a car would cost twice as much as a normally safe car. It may turn out to be impossible at *any* cost. Perhaps we have to change the need: reduce the required speed of safe impact to 10 miles an hour, because then it only increases the cost of the car by 15%.

Thus, in a very basic way, every project is about reconciling the fundamental polarity between the world of What-Is-Needed and the world of What-Can-Be-

Built. These two worlds correspond to the upper and lower halves of the MCC elevator study diagrams (Figure 5). The upper half, being about understanding the problem, is focused out in the world on a specific client or user or market. There is always someone who has a need or a desire, and the task in the Problem or What-Is-Needed aspect of design is to specify that need. The lower half, being about the Solution, is focused "in the shop" on What-Can-Be-Built—what do we have the resources and skills and tools to actually make, what will it cost, and how long will it take?

As you can see, there is an immense difference between these two worlds. When an individual does design, she stands with a foot in each world. Moving back and forth between the two worlds, she tries to create a solution that joins the two polarities of design in an elegant way. Design teams have a bigger challenge. While it is possible for each person on a project team to be standing in both worlds, the tendency is for the polarity of design to be reflected in a polarity of roles. Typically, the world of What-Is-Needed is the domain of the Marketing and Sales department, and sometimes upper Management, whereas the world of What-Can-Be-Built is territory that belongs to the Engineering department (or Manufacturing, Software Development, IT, etc.).

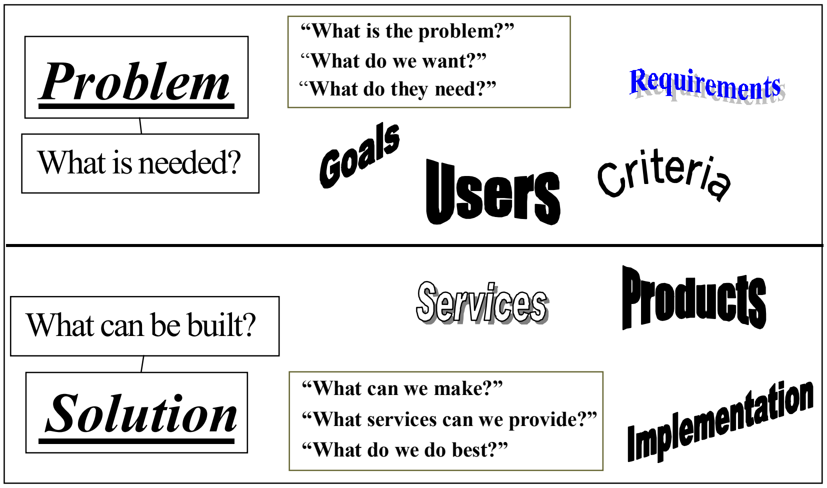

Figure 5: The two parts of the world of design

The inherent unity of the design process turns into a battle between departments. The world of What-Is-Needed, claimed by the Marketing department, becomes a self-referential world with its own culture and customs and language. The world of What-Can-Be-Built is claimed by the technologists, the nerds and hackers who actually build things, with its own culture and customs and language. When they sit down together on a project, the polarity of design stakeholders turns into an inter-cultural war that is expensive and wasteful and ineffective, a war that is a frequent theme in Scott Adams "Dilbert" cartoons.

Thus social complexity is not just a function of the *number* of stakeholders—it rears its head in even the simplest of social networks. Social complexity is also a function of structural relationships among the stakeholders. While large projects have an increasing number and diversity of stakeholders, it only takes one player from each side of the polarity of design—one from Marketing and one from Engineering—to cause the collaborative gears to grind to a halt.

Technical Complexity

The icing on the cake of the wicked problem challenge is technical complexity—the number of technologies that are involved in projects, the immense number of possible interactions among them, and the rate of technical change. For example, to be a serious player in the software industry today, your software must run on a variety of types of computers. Each type (or "platform") has several operating systems, and each operating system has many versions which are currently in the field and must be supported. You must choose what language your software will be written in: Java, C, C++, Cobol, Fortran, etc.

Each of these programming languages has a variety of supported versions (compilers); for example, Microsoft and Sun each have a major version of the popular Java language used in World Wide Web applications. Then you must choose the set of utilities (“library”) you will use for creating your user interface. There are dozens of other choices, and all of these options interact with each other. Moreover, the field is changing so fast that new options become available, and others drop into oblivion, almost every day.

As much as technical complexity raises the risk of project failure, it is also the most well-recognized source of project complexity. So much has been written about technical complexity and how to deal with it, so many tools and methods are available, that there is little to add here. The Dialog Mapping approach [presented in the book] excels at dealing with complex technical information, but the real power of Dialog Mapping is to provide an approach and a set of tools for dealing with the non-technical aspects of project work: wicked problem dynamics and social complexity.

Fragmentation and Coherence

We have described wicked problems and social complexity as the barriers to successful collaboration on projects, thus as barriers to successful projects. It is important to recognize that these phenomena are not something “bad” for which we need a fix of some kind. These phenomena are more like entropy or gravity—they just *are*. Like entropy and gravity, life is much easier and saner when you recognize them and take account of them. They are part of the “physics” of projects. There is no quick fix for the phenomenon of wicked problems, and no list of “Seven Steps to Crush Social Complexity.” In fact, problem wickedness and social complexity contribute to a *condition* that has a profound impact on individuals, companies, and governments.

In my consulting and facilitation experience I have observed this condition in organizations and on project teams. I have seen this widespread and corrosive condition manifesting in many different forms, sometimes as outright panic, sometimes as plodding determination, sometimes as a vague sense of futility. This condition of organizational pain is so chronic, however, that like low-grade back pain, it has faded into the background of organizational experience and is taken for granted and assumed to be normal and inevitable. The condition is not wicked problems, nor social complexity—these are causes of the condition. Once this chronic condition is seen and understood, in my experience, a whole new perspective about work and life opens up.

At its root the condition is *fragmentation*. Wicked problems fragment the process of project work, especially when the problem is misdiagnosed as tame. Wicked problems also fragment direction and mission—if you can’t agree on what the problem is, how can you be aligned on a solution? Social complexity fragments team identity—the ideal of team unity is compromised by the dynamics of competing interests and hidden agendas. The duality of design tends to divide allegiances between requirements and implementation. Social complexity also fragments meaning—key terms and concepts are used in different ways by the different stakeholders. Project teams are often geographically distributed, further fragmenting relationships and communications. Participants in a modern project team are pulled in a thousand different directions by the centrifugal forces of wicked problems, social complexity, and technical complexity.

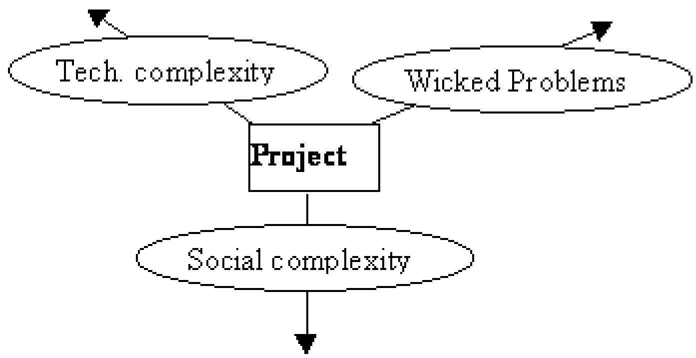

Figure 6: The "centrifugal" fragmenting forces on a project

How can this condition be so pervasive and yet relatively undiagnosed? Why are there so few tools for coherence in the face of fragmentation? One reason is that the fire of wicked problems and social complexity is obscured by the smoke of technical complexity. The complexity and urgency that most project teams face is relatively new, and our culture has been obsessed with technology. Also, despite its obvious weaknesses, the linear “waterfall” model of project management remains the dominant paradigm in the field[7]. Project managers prefer to use the inadequate waterfall model rather than to face the jagged-line chaos of wicked problem dynamics with no tool at all!

The notion of fragmentation is pretty abstract. Because it points deep into the culture and practices of project work, it is difficult to observe directly. There is, however, a more observable indicator of fragmentation: *blame*. Instead of seeing the systemic nature of project challenges and the value of social diversity, we tend to see a big mess, to view it as the result of incompetence, and to blame each other for it. We blame upper management for sending mixed signals or for lack of direction. We blame HR for poor hiring practices and lack of training. We blame the “bean counters” for over-tight budgets and lack of fiscal flexibility. We blame IT for confusion and the lack of stable infrastructure. We blame our customers for not knowing what they really want. We blame each other because we have different personalities and learning styles.

How many conversations do you notice in your organization that involve placing blame?

In times of stress the natural human tendency is to find fault with someone else. We tend to take the problem personally, at an organizational level, and assume that the chaos we see is a result of incompetence or, worse, insincere leadership. Since our education and experience have prepared us to see and solve tame problems, wicked problems sneak up on us and create chaos. Without understanding the cause, there is finger-pointing instead of learning.

If we step back and take a systemic view, we can see that the issue is not whose fault the mess is—the issue is simply failure to recognize the recurring and inevitable dynamics of the mess. The real challenge is to turn away from blame and away from easy technical fixes, and to look in the social domain—in building capacity to collaborate effectively on wicked problems. But, like the drunk who looks for his lost keys under the street light because that's where the light is good, many managers tend to focus on more, better, and different ways of dealing with technical complexity. Or we try superficial and reactionary fixes to social issues. We rearrange the org chart to address fragmentation of corporate identity and direction, which only causes more fragmentation.

As Rittel said, "We are now sensitized to the waves of repercussions generated by a problem-solving decision directed to any one node in the network, and we are no longer surprised to find it inducing problems of greater severity at some other node."[8]

The antidote for fragmentation is coherence. How, then, do we create coherence? In organizations and project teams—in situations where collaboration is the life blood of success—coherence amounts to shared understanding and shared commitment. Shared understanding of meaning and context, and of the dimensions and issues in the problem; shared commitment to the processes of project work and to the emergent solution matrix.

Coherence means that stakeholders have shared meaning for key terms and concepts, that they are clear about their role in the effort, that together they have a shared understanding of the background for the project and what the issues are, and that they have a shared commitment to how the project will reach its objectives and achieve success. Coherence means that the project team understands and is aligned with the goals of the project and how to reach them. Coherence means that a wicked problem is recognized as such, and appropriate tools and processes are constantly used to "defragment" the project. Coherence means that despite social complexity there is a sense of ability and confidence in crafting shared understanding and negotiating shared meaning.

Summary

This paper has been about laying a foundation which identifies the "problem" which Dialog Mapping addresses. This problem is:

- The powerful fragmenting force of wicked problems, compounded by social and technical complexity,
- The confusion, chaos, and blame created by failing to distinguish wicked problems and social complexity,
- The general lack of tools, techniques, and processes for "defragmenting" projects.

The upcoming book will complete the foundation by exploring the impact of fragmentation on the most intensely collaborative event in a project: project meetings. This will set the stage for introducing a medium for creating coherence: *shared display*. Shared display, combined with the IBIS structure and a collection of facilitation techniques, make up the Dialog Mapping technique the book offers for creating coherence (i.e. shared understanding and shared commitment) in the face of technical complexity, wicked problems, and social complexity.

Dialog Mapping is not a silver bullet or a cure-all for the condition of fragmentation. Given the intensely fragmenting forces of wicked problems and social complexity, no single approach will ever be sufficient to create coherence. For knowledge organizations and project teams the challenge is to gain "literacy" in dealing with fragmentation, a set of tools, practices, and languages for crafting shared understanding and shared commitment—quickly, practically, and efficiently. Dialog Mapping is a first step toward that kind of literacy.

Acknowledgements

Many people have contributed to this paper over its years of evolution. I am immensely grateful to Horst Rittel for his groundbreaking work in this area—this paper is dedicated to him. William Weil co-authored earlier versions of this paper, and his contribution still shines through in the current version.

Many thanks to Joan Rostad of Paragon Coaching for her editing and layout contribution to this paper.

References

Argyris, Chris, and Donald Schön, *Organizational Learning II: Theory, Method, and Practice*, Addison-Wesley, Reading, Massachusetts.

Guindon, Raymonde (1990) Designing the Design Process: Exploiting Opportunistic Thoughts. Human-Computer Interaction, Vol. 5, pp. 305-344.

Kunz, Werner, and Horst Rittel (1970) "Issues as Elements of Information Systems," Working Paper 131, The Institute of Urban and Regional Development, University of California, Berkeley, California, Tel: (510) 642-4874, email: iurd@uclink.berkeley.edu.

Rittel, Horst (1969) "Reflections on the Scientific and Political Significance of Decision Theory," Working Paper 115, The Institute of Urban and Regional Development, University of California, Berkeley, California, Tel: (510) 642-4874, email: iurd@uclink.berkeley.edu.

Rittel, Horst (1972) "On the Planning Crisis: Systems Analysis of the 'First and Second Generations'," Bedriftsøkonomen, Nr. 8. Also Reprint No. 107, The Institute of Urban and Regional Development, University of California, Berkeley, California, Tel: (510) 642-4874, email: iurd@uclink.berkeley.edu.

Rittel, Horst (1972) "Structure and Usefulness of Planning Information Systems," Bedriftsøkonomen, Nr. 8. Also Reprint No. 108, The Institute of Urban and Regional Development, University of California, Berkeley, California, Tel: (510) 642-4874, email: iurd@uclink.berkeley.edu.

Rittel, Horst and Melvin Webber (1973) "Dilemmas in a General Theory of Planning," *Policy Sciences* 4, Elsevier Scientific Publishing, Amsterdam, pp. 155-159. Also Reprint No. 86, The Institute of Urban and Regional Development, University of California, Berkeley, California, Tel: (510) 642-4874, email: iurd@uclink.berkeley.edu.

Rittel, Horst, and Douglas Noble (1989) "Issue-Based Information Systems for Design," Working Paper 492, The Institute of Urban and Regional Development, University of California, Berkeley, California, Tel: (510) 642-4874, email: iurd@uclink.berkeley.edu.

Simon, Herbert A. (1969) *The Sciences of the Artificial,* Second Edition, MIT Press, Cambridge, Mass.

jeff@cognexus.org
CogNexus Institute
http://www.cognexus.org

End Notes

[1] See (Guindon, 1990).
[2] See the bibliography for references to Rittel's papers on wicked problems.
[3] Actually, Rittel had a more exhaustive list of criteria for wicked problems. I have attempted to simplify the concept somewhat without loosing its essence.
[4] Wicked problems pose a terminology dilemma. There is no "the problem" in the traditional sense – like Heisenberg's elementary particles, getting close enough to the problem to see it … changes it. Similarly, no "solution" is ever achieved, in the traditional sense. We might better use terms like "domain of concerns and needs" for "problem", and "domain of resolution and satisfaction" for "solution." However, for expediency and clarity, we will use the more familiar terms. So when we say "… developed a solution" we don't mean a real, final, mutually-acceptable solution, we simply mean a proposal that might resolve some part or aspect of a wicked problem.
[5] Indeed, in most government contracts taming the problem has been refined to a high art: the problem definition, as described in the requirements document, becomes a contract which cannot legally be changed.
[6] Design: the approach that engineering (and some other) disciplines use to specify how to create or do something. A successful design must satisfy a (perhaps informal) "functional specification" (do what it was designed to do); conforms to the limitations of the target medium (it is possible to implement); meets implicit or explicit requirements on performance and resource usage (it is efficient enough). A design may also have to satisfy restrictions on the design process itself, such as its length or cost, or the tools available for doing the design. In the software life-cycle (waterfall), design follows requirements analysis and is followed by implementation. ["Object-Oriented Analysis and Design with Applications", 2nd ed., Grady Booch].
[7] There is a recognition of this weakness, and a whole out-cropping of this awareness with new methodologies such as Rapid Application Development (RAD); Joint Application Development (JAD); and Iterative Development.
[8] (Horst and Webber, 1973) See also "Disturbing Complexity" chapter of *Surfing the Edge of Chaos*, esp. the Lapp and reindeer story, p. 161, and the coyote control story, p. 151.

jeff@cognexus.org
CogNexus Institute
http://www.cognexus.org