GOOD COMPUTING: A VIRTUE APPROACH TO COMPUTER ETHICS

A book under contract with
Jones & Bartlett Publishers
40 Tall Pine Drive
Sudbury, MA 01776
http://www.jbpub.com/

DRAFT FOR
June Puerto Rico writing retreat

Chapter 2
Problem Specification
Version 1: 01/30/05 by Chuck Huff
Version 1.1 02/28/05 by Chuck Huff

Based on writings in www.computingcases.org prepared by Chuck Huff and Bill Frey and class documents from University of Mayaguez, PR, FILO 3185 prepared by Bill Frey

In the previous chapter, we introduced the idea of a socio-technical system. Chapter 11 gives you a detailed look at the theory of socio-technical systems and at some methods that have been used to make sure software is designed with the socio-technical system in mind. What we will do in this chapter is introduce you in detail to one way to investigate the characteristics and trajectory of a socio-technical system.

Socio-technical systems have components, embody values, and have a trajectory. We covered the components in chapter 1, and will say a few things about value and trajectories here before moving on to detail a method for investigating them.

Socio-technical systems (STSs) embody values because they are made up, in part by people who have values, but also by rules, laws, procedures, and artifacts that people design in order to support the values they have. It is easy to see how procedures embody values, since we can talk of whether a procedure to order a list potential organ donor recipients is "fair" or "just." If it can be specified clearly, the procedure can even be incorporated into a database system. This would mean that one could talk about the database system as fair or just. In the cases in this book, we will come upon countless places where values become inseparable from the computing systems in which they exist.

STSs also change over time, and this change is not simply random, but has a trajectory (ref). As newer systems replace older ones, as procedures and laws are modified, as organizational structure changes, the STS changes. Those people with power in the STS are often the ones who have the most control over the changes, and thus control over the trajectory of change. New software and computing systems are often a part of organizational change. Thus negotiations about software requirements or about the deployment of the software can effect who has power in the organization and can implement the values of those who "win" the negotiations. Of course the negotiations are rarely couched in terms of power, but instead focus on job descriptions and other such mundane things. But choices of which jobs are automated, and who maintains and has access to the data are value-based choices that affect different people in the organization differently. Thus, when acting as a consultant to identify these choices, their underlying values, and their effects, one can get into difficult political waters quickly, and we provide some advice about this issue later in the chapter.

We have talked a great deal about using social science methods to help understand the socio-technical context of a system. Here, finally, we will present an introduction to how one might actually go about this. Chapter 11 provides examples of several approaches that integrate value considerations with software design. Some of the approaches are prospective, attempting to predict the issues that a particular implementation will face. Others are iteratively incorporated in the design process. Some are focused on maximizing particular values (trust, privacy, democracy, informed consent) while others are designed to make the developer aware of the range of social and ethical issues that a project presents.

But all are based in the assumption that gaining knowledge about the socio-technical system in which a technology is embedded will help the designer to design software that

at least avoids some ethical trouble spots, and perhaps helps to increase the influence of particular values. Almost all of them require that social science methods be used to investigate the socio-technical system.

Socio-Technical Analysis

To do so, we will take you through the process, from beginning to end, of doing a Socio-Technical Analysis on an existing implementation of a computer system. In your class, you may find that this is, in fact, one of the major assignments for the semester. If so, you will get very familiar with this section of the chapter. If not, a quick read of this section is still a good introduction to the sort of socio-technical analysis we have been presenting.

To illustrate the methods, we will use examples from Socio-Technical Analyses that our students have done. Most of these were done as a part of a semester long group project in our classes.

Some overview of the process will get you oriented to the scale of the project. The following are the steps we will take you through:

- *Scoping the Project*: How to select the scope of your analysis.
- *Initial analysis of issues and components*: How to identify the elements of the socio-technical system, their interconnections, and the ethical issues they produce.
- *Data Collection*: How to collect useful data, on time and in budget.
- *Data Analysis*: How to use your data to understand the socio-technical system and to make recommendations.
- *Construction and Testing of Solutions*: How to describe the challenges your project faces, construct solutions to them, and test them for ethical soundness and feasibility.
- *Writing the Report*: How to write a useful report to your client.
- *Facing Problems*: Problems you may face, and suggestions for overcoming them.

You will probably be doing this project as a part of a team assigned to analyze a socio-technical system or one aspect of that system. Some systems are complex enough that three or four teams might profitably spend a term collaborating on its analysis. This may be the first time you will have attempted professional level work for a real client. If so, relax. You will get the guidance you need from your instructor. Your goals for the project are practical: to help the client learn something about the socio-technical system and the social and ethical issues that the client's system presents.

There are two primary goals of an STA, and they flow from the practical nature of the undertaking. An STA is not an attempt at publishable social science research, or a theoretical deconstruction of the meanings inherent in a computer system (though it may *borrow* from both of these approaches). It is a tool to allow you to locate and deal practically with the social and ethical implications of the technology with which your client is working.

One goal is to provide *surprises* about how the system works and the consequences of its operation. By surprises, we mean new information that is not included in the standard story of how the system works. Safety engineers call these *latent errors* (ref), because they lie unnoticed in a system until an accident or critical incident arises. But the idea of latent errors can apply as well to issues of equal access, privacy, property rights, quality of life, or any area where hidden or unnoticed aspects of the system can pose ethical challenges. Thus, the methods used to produce an STA must be apt for uncovering these surprises--simply reading the specification sheets will not do. In addition, as a computer professional working for a real client, you must be aware of the political issues involved in pointing out surprises to designers, managers, and operators of a system. Even if it is your job to do so, one must be politic in pointing out oversights to people you will need to trust to implement fixes.

Another important, practical goal is to give the clients some practice in thinking about the ethical and social aspects of their own system. In a way, the process of thinking about the issues is as important as the product. Even an extended inquiry into a system cannot be sure of turning up all the issues (or even the most important ones). But sensitizing the designers, managers, and operators of a system to the social and ethical issues and giving concrete examples of potential difficulties in the system may make them more aware of the issues and more likely to detect other issues when they arise. One way of doing this sensitizing is to provide clients with a document that will be useful in future modifications of the system, and that will point them to the literature describing the particular problems they face.

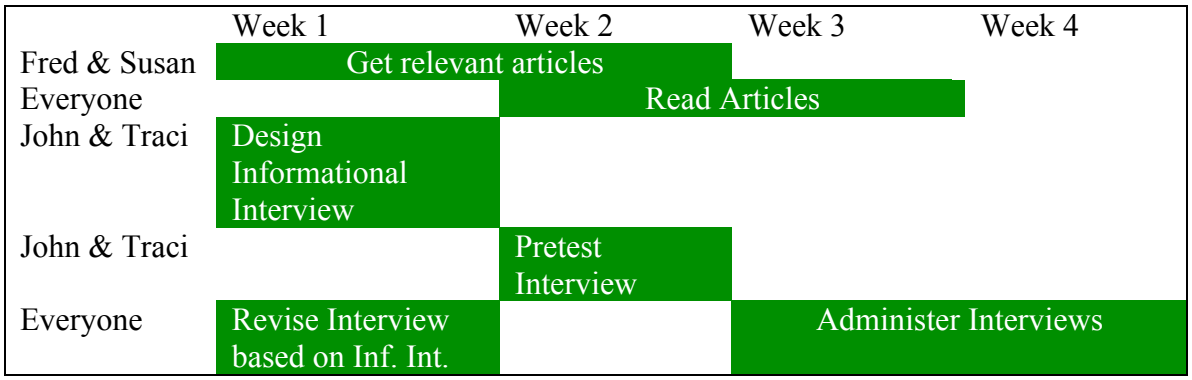
A social scientist might approach a socio-technical system with the purpose of understanding all the nuance and diversity in the system, and would spend considerable effort in a comprehensive undertaking. But your purpose is different. You want to learn something practical about the socio-technical system that will result in recommendations about how the system might be adapted to take into account the social and ethical issues you identify. So, your approach will rigorously practical; you need to come up with the most useful results within the time and budget of the project.

Scoping the Project

If you are doing an STA as a part of a course project, you have at least a clear timeline for its completion: the due date for the project. You will likely have several due dates across the term for the various stages of the project. Keep this timeline fully in mind in your early planning. In a few months' time, a small team working together can learn a great deal about a system. But work with your instructor to keep the task manageable in the allotted time. You want to be sure to attempt enough in your analysis to give useful insight to your client, but not to promise more than you can deliver. This process of outlining the work to be done is called "scoping" a project, that is, determining the proposed extent (scope) of the project.

As with any project you undertake, you will actually need to do a little research at the beginning to help you set the project's scope. Answering these questions will help you begin to scope your project:

- *How large or complex is the system?* You will need some initial information about the socio-technical system to answer this question. You may receive enough from the project assignment you are given, but you may need to do an initial *informational interview* (see the methods section) to obtain enough information. The more complex the system, the more time it will take to analyze. In the end, you may do best by identifying the most critical section of the system and only doing an in-depth analysis on that portion. This is a choice you will need to make early in the project.
- *How many ethical or social issues will be raised by the system?* To answer this question, see the tools in the section below on *Analyzing the Issues*. Your instructor will be a good person to help you in an initial brainstorming of the issues that may arise. You may find that too many issues come up, and again you will have to choose the most critical ones as the focus of your project.
- *How much time will you have on-site to actually collect data?* Some sites will have more limited access than others, and some clients will have more time to give to your project than others. Respect the time constraints your client has. This may mean that you will only have a set amount of time (e.g. a week) in which to do most of your data collection. If so, you need to limit your scope so you can collect the most important data during the window of opportunity.
- *How time-consuming are the data collection methods you plan to use?* If your system (or your assignment) seems to call for a specific data collection method, then be well aware of the real time it takes to make that method work. Interviews and questionnaires sound easy, but are actually quite hard to prepare for properly.
- *How well do the steps of the process fit into a Gantt Chart with your deadlines?* Once you have answered the questions above, you can then prepare a Gantt chart to see how well you can fit all the tasks into the available time. Learning how a Gantt Chart works will help you immensely with this project. Spreadsheet programs will work nicely for preparing these, though there is special software for the very complex charts that some projects need. The simple idea is that of a chart that tracks in an easy, visual manner all the task, deadlines, task assignments to team members, and task interdependencies for your project. Rows in the chart correspond to specific tasks that need to be done, while columns in the chart represent the units of time for the project (usually, weeks). A bar in one row, running across several columns, indicates when that task is to be started and completed. Some tasks can run in parallel (like library research and questionnaire design) but others need to be sequenced (questionnaire design must be finished before the questionnaires are distributed). Deadlines can be marked by a symbol and labeled. Keep the number of tasks you list on one sheet reasonably low (15-20). If you find you need more than 15-20 tasks, make a master chart for larger groupings of the tasks, and then individual charts for the subtasks. Estimating times for task completion is difficult. Use your best guess and then double it. There are many tricks to Gantt charts that make them useful, but these basics will get you started.



Initial Issue identification

You will want to gather as much information as you can before you even begin with your initial informational interview with your client. What publications, past reports, or news articles are there that might help you understand the extent of the socio-technical system? Is there a web site that offers an organizational overview? What can you find out from informal conversation with others? What social or ethical issues is the client most concerned about? What unanticipated concerns might there be?

Of course, the data collection process is supposed to help you answer these questions, but you cannot enter that process blindly, simply collecting whatever comes to hand. You will need to focus your efforts. So gather what information you can ahead of time, use the categories of the socio-technical system we have listed to see what you are missing. Use the *Identifying and Analyzing Issues* section of this chapter, (especially the initial STS x Issues grid) to begin identifying potential issues on which you might focus your data collection efforts. Later, after you have collected all the data you can, you will go back through that section (and also Chapters 3, 4, & 5) to clarify the important issues, construct solutions to them, and test their ethical and practical soundness.

Data Collection

Now that you have some idea of the socio-technical system and the social and ethical issues involved, it is time to fill out your initial understanding by collecting data on the system. This process, of course, is iterative. You may have needed to do an initial informational interview with the gatekeeper for your organization so you could begin to get an idea of what the system was like and what data you needed to make recommendations. As you collect that data, you may find your picture of the socio-technical system changing, and thus your plan for data collection changing. This is the normal process of discovery.

First, we need to think for a moment about the ethics of data collection itself. There are a few basic principles you need to be aware of as you go around poking your head in odd corners of a stranger's socio-technical system.

1) *You are a guest.* You are probably doing this project because it is a class assignment. Your client has an interest in the project probably because they hope to learn something, but also because they are willing to help support your

education. Thus, by agreeing to be "studied" by you, they are giving you a gift. When you take their time for interviews or observations you should always be aware of this relationship and always be thankful (and express your thanks) to them for their support.

2) *You are in a professional-client relationship* (if only a tentative one). Before working with your client, you should look through chapter 5 on career advice to see what advice we give regarding the relationship of a consultant to a client. By agreeing to be a consultant to your client, you are taking on these responsibilities.
3) *Informed Consent is crucial when getting agreement to collect data.* Look at the section on Value Sensitive Design in chapter 11 to see what the components of informed consent are. You need to make sure, in your initial meeting with the gatekeeper for your client, that you take care of all these points. They need to understand what they will get from you, what it will cost them (in time and bother), and get a clear chance to say "yes" to the project. This is likely to involve two stages. At your initial interview you may not be able to tell them exactly what you plan on doing by way of data collection. But you can arrange a closely following date when you can present them with your plan in its entirety. You might assume the person with whom you have your first meeting is the official person (this is the "competence" criterion) to give you permission to collect data in the organization, but you should check: they may need to check with a superior or their colleagues. You should also follow informed consent guidelines whenever you are collecting data from any individuals in the organization. Part of your informed consent procedure should be letting the client know to whom information about the system will be presented. Will you be doing a class presentation? Will your instructor see your report? Who at the client's site will have access to the report? What will happen to the data after you deliver the report? How secure will the data be when you are keeping it? Finally, remember the "minimal distraction" criterion; you want to cover the points of informed consent, but not have this get so much in the way as to distract from your purpose or from theirs.
4) *Confidentiality and anonymity are important.* In the process of collecting data from people, you may find yourself in possession of sensitive data. Because of this, you should assume that all the data you collect is confidential unless you have explicit permission to share it. Think through some of the issues discussed in chapter 7 on data privacy when preparing for your data collection effort. When you guarantee someone *anonymity*, what you have told them is that their identity will not be connected with their data *in any way*. It is very hard to do an anonymous interview, since an interview usually means you know who the person is as you are collecting the data (or at least their telephone number). Surveys can be anonymous, if you are careful. *Confidentiality* means that you will not disclose their identity to others or connect it with data in a way that others might discover it. For instance, even if you do not say who the black, female person considering retirement is, if you mention these three facts about her, most everyone in an organization will be able to identify her. You have then violated her confidentiality. Some clients may not want to be confidential or anonymous, others may. If you are collecting data in a two-person shop, you are unlikely to be

able maintain any level of confidentiality anyway, and should be clear at the beginning about this.

You may or may not be required by your instructor to have your project reviewed by an ethics committee before you begin data collection. Since you are not engaging in real social science research with the intention of publishing, you will likely not need to. If you do submit your project for review, you will get a fair amount of help from the review board on thinking through the ethics of your data collection, but you will need to allow additional time for this review to take place, and perhaps for you to resubmit your project if it should be turned down the first time.

Interviews

The first bit of data collection you are likely to engage in is an interview. We will talk about two kinds of interviews you are most likely to use: the informational interview and the semi-structured interview. There are a wide range of other interview methods (refs) that you may what to know about if you plan to make interviewing others a part of your career skills, but these two types will do for now.

Some Theory. First some theoretical background about interviewing. In an interview, you are establishing a social relationship with someone in order to allow them to give you information that will help your project (and possibly help them too). This assumes that the sort of information you need is something your interviewee can tell you about. But we now know enough about knowledge in socio-technical system to know this assumption is not always true. Consider the following distinctions from the social science literature:

- *Conscious vs. automatic processes*. Some mental processes are conscious. By this we mean we have access to them, and can report on the process of how we came to the output. If you ask someone how they decide to prioritize different tasks, you may well get a list of important dimensions. These are likely dimensions they consciously sift through when making a decision. But if you ask someone how they know that an applicant is "suspicious" and needs further checking, they may only have an intuitive feel. They do not have access to the automatic mental process that produced the output "be suspicious!" This may not stop them from coming up with a personal theory about why they are suspicious, but it is not based on special access to the process in their head. (see refs)
- *Explicit vs tacit knowledge.* Some knowledge in organizations is explicit. There may be, for instance, explicit rules about how travel vouchers are issued. But some knowledge in organizations is tacit. It is not written down, and in fact people in the organization may not be aware of these norms (thought they follow them) and even if they are aware may resist being explicit about them. Some tacit norms or knowledge only work because they are not explicitly acknowledged. For instance, a work-to-rule strike involves only following the explicit rules, and not acknowledging the tacit ones that really let things get done. The power of this kind of strike is evidence for how important tacit knowledge and norms are in an STS (see refs)

- *Procedural vs. declarative knowledge*. Velcro is not so new that you have forgotten how to tie you shoes. But can you explain, using only words and no gestures or diagrams, how to tie your shoes? Or ride a bicycle? Procedural knowledge is embedded in skills about how things are done. And they are often very difficult to make into declarative statements. Thus someone might be able to show you how she does data entry, but not be able to write it down. This is, of course, one tremendously difficult part of requirements analysis in software design. But you will also see its importance in understanding procedure and norms in an STS.

The shared point of each of these distinctions is that something in an organization may look like it is easy to do, but it may be tremendously difficult to say exactly what the procedure is, in all its variation. As you do your data collection, you will want to be aware of these issues.

The Informational Interview is an initial contact with a person to obtain basic information from them. Depending on how much you already know about what you want to know, this interview may be semi-structured in its format (see below) with a set of established questions and question order you follow. Or it may need to be more open-ended than this if you know little about the client and their system. You will have three goals in this initial interview: 1) to establish a professional-client relationship, 2) to obtain informed consent to the project, and 3) to get the initial information you need about the socio-technical system and the client's perspective on the important issues. Thus, your initial informational interview will consist of introducing yourself and your project to the client, giving them the opportunity to give informed consent to the process, inquiring about the socio-technical system and issues, reminding them what they can expect next, and thanking them.

Your first contact will likely be with the person who has agreed to allow your team to do an STA for them. Thus, they will already know a (very) little about what to expect. You need to be able to explain in a few sentences what the project is about, what they may gain from it, and what it will cost them in time and interruption to allow you to do it. You then should give them a clear opportunity to agree or negotiate agreement (e.g. we can't give you that much time or access, but here is what we can do). You will need to practice this beforehand so you can do it in a way that is inviting, understandable, and brief. You will also need to be ready for some negotiation if it is called for.

When asking about the socio-technical system at this stage, you really want two things: 1) a broad overview of the system and its purposes and 2) information about where you can find information about the system. You may already have a broad overview of the system from your assignment. If you do, a set of questions that confirms your understanding of the system is in order. If you do not, you will need to prepare a set of questions that allows you to find out about the system's general structure and purpose.

Once you have an understanding of the main structure of the system, you can then ask about where you can find more detailed information on the system. How can I find out

how the data is structured and how it is stored? Are there manuals about the system that we can look at? Who in the organization can we interview to find out about how the data is shared (collected, etc.)? Are there records of the system's performance? Is there a place where we can watch people using the system (without being in the way)? Would it be possible to distribute a questionnaire to the users (operators, etc.) of the system? These are just suggestions of the sort of questions you should ask. But you can see that in order to frame your questions, you need some idea of the system already in place and some idea of the methods you intend to use.

When asking about the social and ethical issues, you first want to give them an opportunity to tell you what they want you to look for. Likely, they agreed to the project because they have some question that concerns them. What is it? Listen carefully to the answer, as there may be multiple agendas (some hidden) in it. Do be privately suspicious about whether this is the most important issue, but be publicly (and genuinely) supportive of helping them find an answer to their question. You may also wish to suggest some issues that your preliminary analysis has uncovered and ask for their reaction to those. Again, it is important here to be politic in how you phrase things.

Finally, if possible, you want to end your interview by recruiting the person to help you as much as they can in subsequent stages of the project. You may need to come back to them (or to someone they designate) to ask more questions in the future. Find out if this is acceptable and how you can arrange for it.

A Semi-Structured Interview is a more general-purpose interview whose name is taken from the fact that it is somewhat structured, but divergence from the structure is allowed. In fact, the informational interview we just covered is one kind of semi-structured interview. This approach is best used when you know of a target person who has key information or a stakeholder who has a key perspective you need to understand. The exact questions will vary depending on your purpose (learning about a perspective, learning about a procedure, learning about hardware, etc.). But some general pointers will be helpful.

There are two processes of any interview that are occurring simultaneously: the socio-emotional and the task-oriented. The socio-emotional process is about establishing a relationship with the client and is supportive of your primary task-oriented purpose.

Socio-emotional goals in the interview include:

- *Establishing a professional relationship*. This begins with your first contact with the client or the client's gatekeeper (e.g. secretary). If you do not call when agreed, are not ready with answers to easily anticipated questions, and are not confident in your demeanor, you may make it less likely that the client will give you the access you need to actually help them. You are a professional consultant and you need to be prepared like one.
- *Establish conversational context*. All conversations have agendas, even if it is too excruciating to explicitly negotiate them at the outset. But your dress and

demeanor can establish the context for the interview as professional, cooperative, and friendly.

- *Establish trust.* At your first contact, you will be establishing the credit you *may* need later. As you begin to know more about the STS, you may need to ask questions that could threaten your contact persons "face" or social standing. This is always difficult, but even more difficult if you do not have some credit already built up in your relationship.

Task Oriented purposes of the interview may include:

- *Understanding the client's ethical concerns.* Your client may have explicit language (e.g. HIPPA or FERPA privacy laws in the US) to articulate their concerns. But they may also have concerns that are unvoiced, implicit, or unrecognized. You may want to help your client find language to describe these, or come to discover a new concern of which they were unaware.
- *Discovering the objects and procedures in the system.* You may simply want to get the best outline you can of the STS. This may involve drawing pictures, looking at organizational charts, asking for handbooks, or discussing procedures for special cases or times.
- *Discovering variation in the system.* An important thing to recognize about an STS is that the simple story is often complicated by many special cases. What are the special cases?
- Discovering power relationships. This is a tricky issue to ask about, but if done right can be quite revealing. Who approves what? How does money and time get authorized? Who is left out of this decision making? You often do not need to ask this question directly, but can note whose name does not come up (which requires knowing all the roles in the STS).
- *Recruiting the informant to help.* You will need cooperation from your client in order to get the information you (and they) need. So one of your golas will always be to cultivate and maintain your relationship.

There is no single best interview approach to do all these things. For outlining and STS, a more formal interview with lists and drawing tasks will be required. For establishing trust and recruiting help, an informal, conversational tone may be best. And all these decisions need to be informed by the organizational culture. Is it hierarchical? Informal? You approach will need to differ to match the expectations of the client's STS.

Interview Structure. All interviews should begin with some setting of the context. As explained above, there is both a socio-emotional context and a task-oriented one. It is important to achieve some clarity on these at the beginning. Informed consent information is an important piece of context, as is your *specific purpose* in interviewing *them* at *this time*. Explain all three.

In structuring an interview there are three different options, though they all involve moving systematically through the issues:

- Small to large. You might start by asking specific questions about particular items your want to know. Starting small in this way allows their answer to be less

influenced by whatever "large picture" information you might have otherwise provided. To do this, you will need to start by begging their indulgence at asking quite specific questions without context, and you can mention the need to avoid bias as a reason. But often this is an uncomfortable interview for the the interviewee, since they are being asked to trust you that all will be made clear later and that their answers will not make them look stupid or naïve. Thus you should only use this if you feel it is necessary, and then only after establishing some trust with the client. A final disadvantage of this approach is that it does not allow the client the context he or she might need to be of the most help. There might in fact be a procedures manual that they do not get to tell you about.

- *Large to small.* This organizational approach gives the context first, then asks for the cooperation of the client in filling in the details. The advantage *and* difficulty here is that by making the context explicit, you invite the client to tailor their responses to the context.
- *Grand Tour.* This involves a structure that goes from place to place in the organization or STS. For instance, you might ask questions about each of the components of the STS. This only works when there is in fact a structure to the questions that need to be answered. It is helpful to give an overview of the structure at the beginning of the interview so the client knows what to expect.

As you finish the interview, you want to end with some opened ended questions that allow the client to give you the larger picture (even if you have already asked for it in the interview). Questions like "What important things have I left out?" "What didn'd I ask you that I should have?" are useful final questions in case there is a lingering issues that has been unaddressed. Again, you will need to fit the phrasing to the kind of culture in which you find yourself.

Finally, at the end of the interview, thank them, tell them what will happen next, and thank them again. They will want to know how their data will be used, when you will have some results, when you might next contact them etc. All this shuld be covered before the final thank you.

You should carefully consider the ordering of the questions. As a rule, you should move from the general to the specific, and finish up with a "clean up" question like "Is there anything else I need to know about X (the system, your perspective) before I go?" Remember that your interview is a conversation, and that the interviewee will use information from previous questions to help them understand what you want from them. You can use this to advantage (for example, by asking for their general thought about the system first, and thus not letting them know which pieces concern you most) or you can be tripped up by (for example, by mentioning a particular issue too early and having it color the rest of the interview). A pretest of the interview can help you identify these problems.

Be aware of two pitfalls in question design: dual questions and leading questions. *Dual questions* are actually asking two questions at one time. Some times these work (Do you like chocolate ice cream, and if so, why?) and this is usually because there is an important

dependency in the link between the two questions. But sometimes these are simply confusing (Do you think the system protects the users' privacy or is it badly designed?). Again, pretesting will help you figure out these problems.

Leading questions make the interviewee think that you favor a particular answer. You may think the system invades their privacy, but if you want to know if *they* think so, you should not ask a question like "Do you think the fact that the system makes your email publicly available to anyone who wants to abuse it is a problem?" Besides being confusing, this question is likely to make the interviewee think that perhaps they *should* be concerned even if they are not. You might instead ask them about a series of functions about the system, some you think privacy invasive, and some not, and ask them what concerns they have about each of these.

When asking people to remember things, ask for specific, recent, short time frames. For rare items, a time frame might be the last year, but for more frequent occurrences, (e.g. number of telephone calls) the last day might be the right time frame. Asking for how many calls they get "on average" requires an estimate. It is better to ask over a series of days how many calls they got each day, and then to ask if these days were unusual.

For all interviews, you should pretest your questions on a colleague to be sure someone can understand them as you have written them. This also gives you the advantage of practicing the interview. Obviously, your practice partner won't be able to answer the questions, but they can reply to each question by summarizing what they think you asked for, and commenting on your performance at the end of the interview. They can comment on your delivery but also on whether they think the questions you have included are appropriate or likely to be confusing, leading, irritating, or intrusive.

Equipment for interviews: Digital audio recorders are now quite reliable and reasonably cheap, so you can record your interview if you first get permission from the interviewee. You can later take more extensive notes while listening to the recording. To make sure there is informed consent, you should be sure to tell them what you will do with the recording, and how it will be protected and disposed of after the project. But you should be aware that a recorder can subtly change the dynamics of an interview. Many people become more careful of what they say when they know they are being recorded. Transcription of the entire interview will likely be a waste of time. At most, you might want to transcribe key sentences or phrases to get the specific wording of important ideas or feelings. An alternative to recording is a carefully designed check sheet for your interview that allows you to read the questions and take notes on the answers in a straightforward manner. Again, you should tell interviewees what you plan to do with the notes during and after the project.

Interviews can also be done over the telephone or even over email. Remember that you lose spontaneity when doing this, though you gain in flexibility of scheduling. You also lose the ability to really establish a relationship with the person that might help you later when you need to know more or to be given access to other people the person might know.

Interpreting Interviews. People may on occasion simply tell you what they think you want to hear. They will do this not because they are perverse, but because they want to have a successful social interaction with you, be agreeable, and be liked by you. These are all understandable motives. It is important in establishing your rapport with them that you make sure they know what you really want is what they know. People may also give you the "official line" about how things are done but not mention the exceptions to these procedures. They may forget to mention how things change on important days (like the end of the fiscal year, or during rush hour) so you should probe to see if there is variation in any procedures they describe to you. You should also remember that people only can tell you what they *think* is true, and that they may actually *do* things differently than they can explain them to you. You cannot guard against this in an interview, but you can combine interviews with field observation to check for it.

When To Use Interviews: You should use interviews whenever you can identify a small set of specific individuals who have information or perspectives you want to hear. In general, you should prefer interviews to questionnaires, since you get so little feedback from questionnaires about when you are asking the wrong questions or leaving things out. In most situations you will not need to use tightly structured interviews. You should recognize that the more structure you put in an interview, the more it becomes clear to the interviewee that you are controlling the conversation. Since in most cases you want to recruit your informants to collaborate with you in helping you understand the socio-technical system, you should avoid these kinds of power plays. Or you should at the least apologize for them and explain why they are necessary.

Field Observation

Field observation is probably the most fundamental tool in your box of methods to understand a socio-technical system. It involves, in short, going to the site in question, making careful observations, and recording those observations in a useful manner.

Some initial notes about the difficulty of doing good field observation. Some of these methods are borrowed from the fields of sociology and anthropology and are called "ethnomethodology." Professional level ethnomethodology involves immersing yourself in a culture so that you can see the culture and its rules from the inside. It is very difficult to do and very time intensive. If you are part of a very large software organization, there are likely to be some people I nthe Interface Design division who are familiar with ethomethodology. Use the expertise of these folks rather than striking out on your own. But remember that useability testing often concentrates on the individual level (how does this person use the system interface) rather than the socio-technical system. You will be asking social and ethical questions about the larger system rather than cognitive, perceptual, or ergonomic questions about the individuals using the interface. Sometimes these will be the same questions, sometimes they will not.

To get some idea of how difficult it is to do these approaches in the proper depth to generate publishable social science research, consider that the average length to publication of professional ethnographies of work is 8 years (ref), not counting

negotiating initial access (a year or more itself). So don't expect professional quality from a week or two in the field.

When you are doing your observation, remember the difference between observation and interpretation. Be as concrete as possible. Don't say "she was angry" but instead take note of how she expressed herself, what was said, what was done.

What to Observe. You first need to decide what you want to observe. This will give you clues to answer the other questions posed below (like where and when). Your informational interviews and your preliminary analysis of the system should have given you some idea by now as to what parts of the system most deserve some sustained attention. You need to be careful to maintain your attention on those parts of the system. An easy mistake is to go to a site to "look at stuff" and "see what happens there" when you do not have a clear idea of precisely what process, resource, system, or outcome you want to observe. Some preliminary scoping of a site in this manner is not bad, but you will spend your time much more wisely by outlining at the beginning what you hope to cover.

To begin your list of things you want to observe, look at the parts of the socio-technical system you established in your preliminary analysis. These parts (taken from the list in chapter 1) should give you an idea of the particular people, processes, or other things you want to observe. Here is another list suggestive of some of the most likely things you will want to observe"

- *Physical Settings.* What does the physical setting allow actors to do or constrain them from doing? For example, a secretary's office is often interposed so that it is the only access to a manager. This physical setup restricts some actors and frees others.
- *Activities*. What activities do the actors engage in? Your informational interviews may lead you to think the actors primarily do one thing, but watching them at their jobs may help you to find things they forgot to mention, or find that a minor aspect actually takes up more time than was thought.
- *Social Environment.* This is the environment that other people create for each other. Is it competitive? Friendly? Informal? What specific things can you describe to make the case for this?
- *Formal Interactions.* These are those interactions that are officially a part of the job of the individuals. Who initiates them? How long do they last? Where do they occur? Who decides when they are over? What occurs before, during, and right after them?
- *Informal Interactions*. These are all those social niceties that help people to maintain relationships with their coworkers. They may also serve other purposes, such as emphasizing friendship after a tense meeting.
- *Non-verbal communication.* People constantly move about in their environment in a way that gives information to others. They point, nod, grunt, mutter, slouch, stand up, sit down, cross their arms, look at each other and away, touch various

parts of their bodies, and handle various artifacts (either instrumentally or toying with them). How do these interactions guide the action on the site?
- *What does not happen?* If you know from an informational interview that something is supposed to happen when a customer comes in the door, look to see if it does. Also look to see who does not talk to whom and what people avoid doing.

The basic idea is to look for a small set of things that you want to observe. Except on a preliminary scoping observational trip, you will want to confine your attention to a few items (3-5) for each observational trip. More than that and you will not be able to watch carefully for the things you care the most about. If you have a great many things to look for, you should first consider cutting your list (and the scope of your inquiry) down. Next you should consider making multiple trips to observe different sets of things on different trips.

One more take on what to look for. There are three different questions to ask that have a great deal of social science research to support their utility (see refs for reviews). Each of these are somewhat contrarian in their view. That is, they ask what is *really* happening rather than what people say is happening.

- *How are routine jobs complex?* You might think of this as "making the familiar strange." Some jobs, like data entry or table servers, are low status and often described by both their occupants and their supervisors as "routine." When you hear this, be (politely) suspicious. You might ask how this routine jobs changes when there are mechanical failures, bottlenecks, speedups, or other time pressure requiring shortcuts. What happens when materials (e.g. forms, raw materials) are defective? How are value tradeoffs resolved by the way the job is structured. For instance, quality can be traded off for efficiency, and control can be traded off for service. Do actors in routine jobs have any control over how these tradeoffs are negotiated? Sometime we say a job has been "deskilled" when in fact the locus of expertise and control has simply been shifted (to another part of the job or to another routine job) (see refs).
- *How are complex jobs routine?* You might think of this as "making the strange familiar." Many complex jobs (doctors, lawyers, engineers) claim that they use complex, nuanced, intuitive judgment to resolve problems. But often close observation can show that the supposed complexity can really be captured with a simple model (refs). In an emergency room, there are procedures that allow personnel to make crises routine. Other complex jobs also have these simplifying structures. In jobs that require personal interaction (e.g. counseling sessions) the personal is often structured to make it impersonal and thus less intrusive (ref). Complexity of often routinized to make it manageable. You can find places where this happens by looking for where there are limited resources (time, money, emotion, etc.). Identify these and look for how they structure the interactions. (see refs.)
- *Who has power and resources?* You can think of this as "who is *really* in charge here?" Officially, a manager may have authority over almost every policy in an

office. But it may actually be the secretary (who has been there through three company names changes) who makes the rules, because she or he makes the exceptions. Does power correlate with gender, race, income, organizational status, etc. Don't expect that it will, but look carefully to see. Often power in organizations if diffused even if the organizational chart is hierarchical. (See refs.)

Where and when to observe. These questions are mostly answered by knowing precisely what you want to observe. But a few tips here will help.

- *Use triangulation.* What you learn in your interviews is what people can put into words. You can use your observation to confirm (or disconfirm) what interviews lead you to expect. Follow up interviews can then confirm or help you reinterpret what you have observed. Observations taken at different times, sites, or vantage points can support or disconfirm each other.
- *Look for variation*. Things rarely occur the same way every time. Look for variation in your informational interviews and then seek to confirm that variation in your observation. Think about situations where unusual (and interesting) things might happen and then look to see if they do.
- *Be creative*. Do not assume that the office where the system is installed is the only site you need to look at. Is the data collected elsewhere? Is it handled elsewhere or sent elsewhere? Remember to consider the whole socio-technical system as potential fodder for your analysis.
- *Be systematic.* If you have the resources, confirm your observations by looking at more than one time and more than one site. Choose the times of day and vantage points with an eye to covering what you think will be the most important things.

How to observe.
You will need to decide if your observation will need to be interactive or unobtrusive. An interactive observation involves watching a person or group as they do a task and interacting with them about what they are doing. Unobtrusive observation involves watching from a respectful distance and only taking notes on what you can see.

Interactive observation. A standard way of doing this is called the think-aloud protocol (ref). You watch what someone is doing and they have instruction to keep a constant commentary running about their action. What they comment on is up to you (with, of course, their consent). You might ask them to simply describe their actions. Or you might ask them to describe their reasons for their action. Or you might ask them to describe what they are thinking about. Other than a note taker, your job is not really to interact with them except to prompt them to continue their running commentary. Another way of doing this involves asking a series of targeted questions during the task. This can be very informal as an introduction to the data entry protocol at a site, or it can be highly structured. An important issue on both of these kinds of interactive observation is to not overwhelm the person with too many tasks. Trying to report what they are doing, what they are thinking, their reasons for their actions, and their feelings all at once, while doing the task, is simply impossible. Less is more here; you will get better and more complete

information if you ask for less. You might structure to task, asking for the person to do a specific set of things, and then see how they are done, noting the resources used and ethical or social issues that arice.

Unobtrusive observation. This kind of observation should be just what it says: unobtrusive. Ideally, the people at the site should get used to your presence so that they do not really notice you there. You can't really hide in the woodwork or install secret cameras, but can dress so you are unobtrusive, stand out of the way (but with good sight lines), and avoid contact with people.

How to take notes. A critical eye and careful note-taking are at the heart of good observational research. Your field notes are your record or what you see and think. They contain two parts: the description and the interpretation. You might separate these into two columns, or keep them in two different files. But it is crucial that you keep them separate from each other and recognize the difference.

The description is just what it says. Descriptive. If you think a room is friendly and inviting, that is not description. What is it about the room? Lighting? Color? Use of space? Furniture? Attitude of the person in the room? How do they show that attitude? Smile? Direct their gaze at you? Speak a welcome?

The interpretation is what goes on in your head as you try to make sense of what you observe. It can be simply labeling an interaction as friendly. But it might also be a speculation about why the interaction was occurring. Or it could be a question about what purpose the layout of the room serves. Here is where you let yourself be creative in thinking about how the pieces of the socio-technical system fit together and what ethical and social issues they raise.

Your notes can contain pictures you draw, flow diagrams, short phrases with arrows between them, short quotes taken from the actors at the site, and even pieces of paper or other documentation that you might find on the site (with, of course, permission). You should be as quick and concise as you can in taking notes. You will have time to unpack and interpret your notes later.

But if you notice it, you should take a note. You will simply not remember the next day. This means you will have a lot of notes you don't use in the final report. That means you took good notes. Often two thirds or more of notes go unused. But you can't predict *which* two thirds until afterwards.

What to do after observing. The first thing to do after an observational bout is to consolidate your notes. Do not wait for a convenient time to do this. Do it now. The longer you wait, the more you will forget. After 45 minutes you will likely have no idea what "Fred rep. 4" meant, though it must have been important enough to write down. Write out in full sentences what your cryptic notes mean. Now. Also, sort out interpretation and description at this time, while your memory of the situation is fresh. You might discover that your description says "Susan was angry." Right after your

observation you might be able to recognize this as interpretation and then write a real description of what Susan did to make you think she was angry. You should also take the time to record carefully the circumstances of your observation (e.g. time, date, people present, site description, etc.). You will not remember these details later. Finally, you should take some time to reflect on the ethical and social issues that your observation brings to mind. This may be best done after a little while, and with another team member with whom you go over the notes. But if you think of it, note it then. Later you may only be able to remember, "Gosh! I had a really neat idea, but it is gone now."

Questionnaires

Questionnaires are often the tool of first resort for those who want to collect data about how things are done in an organization. This is usually a bad idea. Good questionnaires are notoriously difficult to make, take up the time of a large number of people answering them, and often do not give the information the designers hoped for. For this reason we recommend not using questionnaire unless you have done all the groundwork so they can be short, to the point, and only given to those from whom you really need data. Above all, they must be pretested before being used.

For the purposes of doing an STA, you want to use a questionnaire if you have a relatively small amount of information that you need to get from a large number (more than 12 or so) people. If you just need the information from 12 people, you can likely simply ask them on site. But if you need it from a set of people who are geographically dispersed or from a large number of people, the questionnaire is the way to go.

Some Questionnaire Design Background

First, some reminders about what we covered in the section on interviews. People who answer your questionnaire are not just providing you with information, they are also engaging in a social interaction with you (yes, you). They want to represent themselves as competent, moral, helpful, people in their organization. These self presentational issues can produce both bias and error (see below). They may misunderstand your question but be too polite to note in the margin that they were confused by it. Remember that answering a question is a communicative, social, and last and often least, cognitive act.

You should also remember from that section that there are things that people may know procedurally, but not really be able to say declaratively. And there may be processes that go on in their heads that produce some output (I distrust our dispatching system) but to which they have no real access. This will mean that their ideas about the source or process behind their intuition is only an idea, and not the result of special access they have to their own mental state. Questions can ask people to tell something, and they will comply, but they may not really know the answer and instead answer only out of politeness.

Second, we need a bit of psychometric (ref) theory. If you assume that your question measures something called the "real" value of the item in question (e.g. number of emails received per week), the actual answers people give you represent:

Score = True Score + Bias + Error

The true score is what we meant above by "real value." Bias is anything that makes responses move in a predictable direction away from the true score. People's desire to look important may introduce bias by inducing them to report more email than they actually receive. Error is randomly distributed around the true score. Simple forgetting may make people mis-report their email, but this may only increase error in the score (and thus the range of the estimates you receive) but not introduce bias.

The distinction between validity and reliability also is important in assessing the usefulness of a questionnaire. Reliability is usually used to measure the extent to which a measure is replicable. If we measure it twice under the same conditions, does the measurement come out the same? Validity is the extent to which a measurement actually measures what you say it does. When you ask people how satisfied they are with their job, what does their answer represent? How they feel today? Their satisfaction with pay? With working conditions? Their likelihood of quitting? All these are aspects of job satisfaction, and if your general question does not correlate with them, then you might worry about its validity.

Questionnaire Design

The primary requirements of your questionnaire are that it should be concise and clear. Concision is achieved by paring down your set of questions until you have the absolute minimum length of questions and number of items. By no means should your questionnaire be more than the front and back of a sheet of paper. Ideally, it should have 3-7 questions and be no longer than a half sheet of paper. There are a range of good reference texts on how to construct good questionnaire items (refs).

Clarity is achieved by pretesting. You can pretest your questions in two ways, and you would be advised to use both methods before committing yourself to contacting people to actually fill out the questionnaire. Get a good proofreader to walk-through of the questionnaire first. In addition to correcting grammar and spelling, they should also try to tell you, after each question, what they think you are asking for and whether they are confused by the question or not. In addition, you might get them to give you feedback on the questionnaire in the same way as was suggested for feedback on interview questions (e.g. looking for biased questions). In ideal circumstances, you should get two or three people to do this for you. None of them should be a part of your team, or acquainted with your project in any detail. You can then move on to more live pretesting. To do this, get two or three of the people with whom you have done informational interviews to fill out the questionnaire (if they have time) and give you feedback on it.

Here are some more tips on questionnaire design:

- Given the error introduced by faulty memory, ask for concrete items (number of non-junk emails) with short, recent time spans and clear, relevant time markers (yesterday).

- Use multiple ways into memory to triangulate and to trigger recall: have them report their behavior, their attitude, check behaviors and attitudes on a list, rank order similar items, etc. If you get differences, ask why. (But beware looking manipulative by being too blatant in asking the same question twice. Instead ask the same question with slight differences in emphasis).
- Avoid ambiguity (double questions, word choice, phrasing). Pretesting is the best way to check for this.
- Avoid large words. Again, pretesting can help you find words people don't understand.
- Be aware of question order effects. You might want to ask some questions earlier to set the stage for later ones. But do this intentionally rather than accidentally.
- Make your questionnaire short, or your respondents highly motivated. They more higly motivated you respondents, the longer your questionnaire can be.
- Collect numeric data when possible: Likert scales (e.g. 1 to 7 with labels for the ends and middle) or summed dichotomies rather than yes/no (they allow you to look for things like reliability and bias, plus they allow more powerful statistics). Again, it is best to work with a professional when designing such things.
- Did we mention the importance of pretesting? It really is important. Do it. More than once.

A final bit of advice about questionnaire content. It is always best to have an open-ended "clean-up" question at the end of a questionnaire that says something like: "Are there any comments you would like to make about this questionnaire or this topic?" You should then leave space for them to write in any comments.

When handing out questionnaires, you can email them, print and distribute them by hand, send them by company mail, or direct people to a website. All these methods work well, but they all should be preceded by a cover letter explaining the purpose of the questionnaire, what you will do with the data, etc. (see the section on informed consent).

Archival and physical trace methods

A final type of method we should cover is archival or physical trace methods. Archival methods involve looking for documents or artifacts stored in archives, libraries, storerooms, file cabinets etc. that will help you understand the socio-technical system. Many systems have manuals associated with them. These often contain a reasonable amount of information on how the system works from the perspective of the computer system. Often a special purpose piece of software will have the original requirements analysis or other design documents available. These provide a great deal of information about what the computer system was designed to do. They can, however, be misleading with regard to how the system is actually being used (or even how it actually works), so claims made in these documents need to be confirmed by interviews, observations, or other methods. Architectural drawing can give you a good idea of the physical layout of an area and some idea of the original purpose of the rooms.

Physical trace methods give you information in a sort of Sherlock Holmes manner. For instance, on an older computer, you can see which keys are used the most by which ones

have the letters rubbed off or dirty. Looking at the sort of cheat sheets, tip sheets, and notes that people accumulate around their desks where they are operating a computer can tell you about what they do and what information they need handy. This is a bit of a mix of a physical trace and archive. Where the carpet is worn tells you where people walk.

Choosing among the methods

You are best to use several methods rather than to rely on just one. The name of the game here is triangulation. If the manual says something is done one way, interviews say it is done another and observations show that there are some instances where exceptions to the process are made, then you have learned a great deal about how that process actually works in that organization. Each method has strengths and weaknesses. People can only tell you in questionnaires and interviews what they can articulate. But they can tell you about their feelings, thoughts, plans, and intentions, while observational methods are blind to this kind of information. It is only in special circumstances that you can observe someone's plans. Observational methods can tell you what people really do, compared to what they say they do (or what the manual says they should do). Archival records like requirements documents give you a systematic presentation of the system, but again, they may only represent the plans for the system and not how the system is really used.

So the best approach is a multi-method approach. You can then begin the job of piecing together the story of the socio-technical system from these multiple sources.

Data Analysis

Attempts at data analysis need to be guided by the overall purpose of the project. Remember the original scope of the project you set out and concentrate your data analysis on achieving an understanding of the data that furthers that original purpose. You may come upon surprises in your data collection or analysis that require you to reconsider the original scope. But do this carefully, since given your resource constraints (particularly time) you may need to choose to drop some area of investigation in order to follow up an interesting lead. You should only drop some line of inquiry like this in consultation with your client. You have made a promise to them about the scope of your project and should consult them regarding any significant change in that scope.

Once people have numerical data in front of them, they are often tempted to run this through a statistical program to get all the standard summaries of the data these programs produce. Resist this temptation, and remember the focus of your project. Your job is to use your analysis of the socio-technical system to help your client think systematically about the ethical and social issues that are associated with that system. Producing summary statistics, graphs, tables, and charts with statistical programs is easy, and will not take too much of your time. But the point is not to present these statistics, but instead to present to the client your recommendations regarding the social and ethical issues that are associated with the system. So by all means look at the statistics, but do so in order to see what implications they have for ethical and social issues.

One analysis method we recommend is the construction of life-cycle scenarios. A life-cycle scenario follows an entity (person, object, data) over a targeted period of time, reporting on the actions of or changes to the entity. For example, you might construct a day-in-the-life of a data entry clerk from their entry to the office until they leave. Or you might follow the data of a person from the time it is collected, though data entry, to storage and archival, and eventually to its deletion (if it is ever deleted) from the system. Or you might document the load and repair history of the typical modem in a modem pool across a semester or across its lifetime. To build these life-cycle scenarios you will need to pull together the information you have about the entity to form a comprehensive picture of it across its life cycle (or across some carefully chosen portion of its life cycle). This often requires an iterative cycle of data collection, analysis, and then return to data collection to fill in the gaps in knowledge.

What these life-cycle scenarios provide you is a comprehensive picture of the regular operating procedures, but also the variations of those operating procedures that occur at special times (shift changes, end of semester, yearly reports, power outages). So it is important to carefully choose the targeted period of time over which the scenario is completed. And it is best to know ahead of time you intend to do a life-cycle scenario, so you can collect the relevant data on your early passes of data collection.

Identifying and Analyzing Issues in the STS

Analyzing the issues and components:
How can you begin to identify the socio-technical system, its components, and the social and ethical issues associated with them? We provide here several tools for beginning this process. It is important to emphasize at the beginning that this is one stage where it is easy to get lost in detail. Remember your overall goal, and make sure you spend time looking for the most important practical outcomes from this analysis. You will find yourself iterating through these approaches several times as you do you socio-technical analysis. You will do some initial identification of the issues, and then after using those to guide your data collection, you will be able to do a better job of outlining the issues and really defining them using these tools again.

Identifying components of the system

Chapter 1 and chapter 11 on Social Frameworks list categories for the components of a socio-technical system. Use this list and the initial description of the system to try to identify as many of the components (or classes of components) as you can in an initial brainstorming session. After you have done this, you may feel you have enough to go on, or you may want to do an informational interview with the client to get more detail. Remember that at this stage you are not collecting data on the details of how the system operates (that is the data collection phase), but just trying to get a basic outline of the system that will help you identify potential social and ethical issues. Drawing pictures, flowcharts or other ways of making your understanding visual can be quite helpful here.

Making an initial STS x Issues grid

The framework we present here is useful for initial brainstorming about the ethical and social issues that might arise in a system you are investigating (See Huff & Martin, 1995,

for the source of this framework). The columns represent various potential ethical issues, while the rows represent various levels of social analysis on which those ethical issues might arise. The list of ethical issues in the columns is not comprehensive. You will need to do your own looking at the literature and brainstorming to make your own framework for analysis. You can start this with an informational interview with the client to discover what concerns the client has. This may add a column or two to your framework. But you can also work with your reference librarian to begin turning up the professional literature in your client's industry sector. Medical facilities have specialized journals dedicated to the managerial problems associated with running them. Perusing this literature can often show you what concerns people in your client's field have about their computing systems. A reference librarian can put you in touch with many different search tools to help you discover a literature about the social and ethical issues associated with your client's work. These issues will give you clues about what columns to add to your framework.

		Ethical Issues						
		Quality of Life	Use of Power	Risk and Reliability	Property Rights	Privacy	Equity and Access	Honesty and Deception
Levels of Social Analysis	Operator							
	Data Entry							
	National Regulators							
	Sales Agents							

The rows in the framework will also need to be relabeled for your particular client. There may be several different roles that individuals in the socio-technical system have, and you will want a row for each important one. In addition you will want rows for the larger groups associated with the system. In essence, what you are doing here is a modified form of defining a list of stakeholders, but sorted from the individual level to the level of larger social units. See chapter 4 for tips on how to generate a stakeholder list.

Once you have gotten the rows and columns defined for your framework, you can then set about asking which ethical or social issues arise for each of the various stakeholders. As you do this, you will begin to see your analysis emerging. Again, remember to be relentlessly practical in this analysis. As you think an ethical issue arises, make note of what data you need to make sure that this is the case. Your job in the end is to investigate the actual system as it works and to make recommendations for modifying the system based on this data collection.

The point of this project stage, then, is to identify those aspects of the socio-technical system that need further investigation. For example, is data collection, entry, storage, retrieval, reporting, archiving, or disposal more interesting (or ethically important)? Your

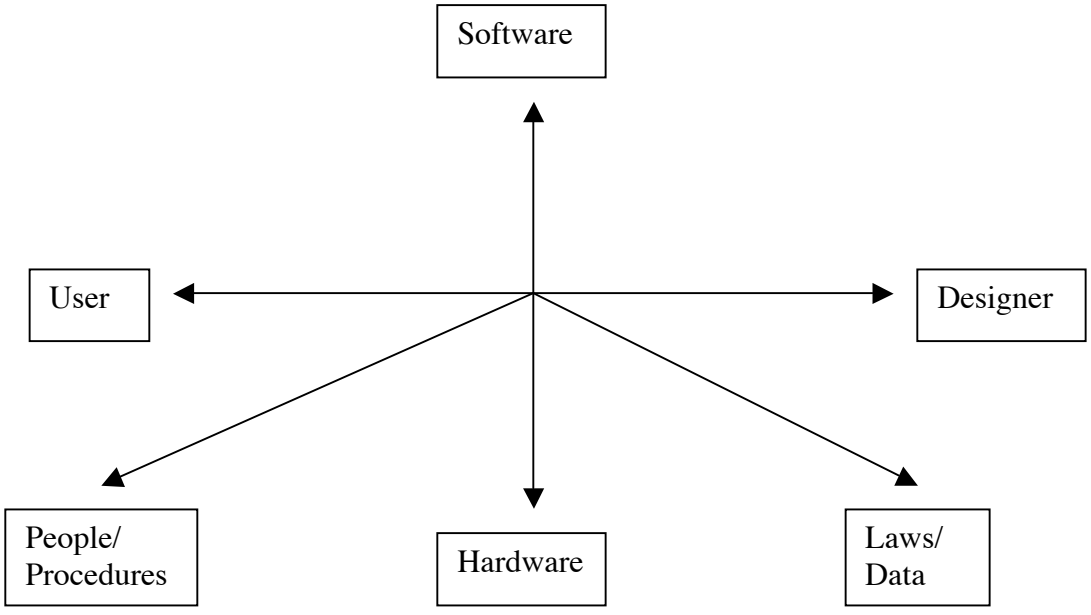
methods will allow you to look at all of these as a system, but only a few of them in any depth.

Mapping design dimensions in a value space

In this book, we are looking in detail at five intermediate moral concepts in computing practice: safety, privacy, intellectual property, free speech, and equity & access. In problem specification, we approach each of these concepts as values that are embodied in the socio-technical system and that can be located on dimensions we might specify in that socio-technical system. For example, safety could be specified in a number of ways but one of special interest to us would be to look at it on a continuum from user responsibility for safety to designer responsibility for safety. (Leveson does this in *Safeware*, ref) This would form the following dimension:

User responsibility for Safety ---------- Designer responsibility for safety

Another dimension could look at whether safety was embodied in the socio-technical system in terms of software/code or in terms of another aspect of the socio-technical system such as hardware, physical surroundings, people/groups, procedures, laws, and data/data structures. The following figure helps to visualize this analysis of embedded value:



For example, in the Therac case, the designer takes primary responsibility for safety by developing software controls that ensure that the turntable is properly aligned (see chapter 6 for details). The operator intervenes only when there is a treatment pause and here only when called on by the unit to decide whether to reenter the treatment data or to override the machine pause. (The error message itself only provides the operator with an indication that a pause has occurred because of a disparity between the treatment

variables entered and the current setting of the machine. But it does not indicate the extent of the misalignment or its cause.) Designer responsibility for safety would locate the Therac system toward the right side of the figure above. Moreover, since the designers have taken responsibility for safety away from the hardware controls of the previous Therac models (6 & 20) and have placed it in the system's software, safety on this dimension in the Therac STS would be located somewhere in the top half of the figure toward the top of the axis bounded by software. We could further elaborate safety embodied in the STS by looking at the legal context (the role of the FDA which required corrective action plan for problems reported) and the role of the different corporate organizations (for example, the responsibility of the manufacturer, AECL, to correct safety problems and report the results to the FDA as well as Therac users). This would help guide us in providing the description of the STS in which the Theract-25 unit was used that would also indicate how the value of safety was embodied in this system.

We could develop similar value figures using free speech (free vs. controlled speech), privacy (transparency vs. privacy/confidentiality), intellectual property (free vs. restricted access), and equity& access (equal access to computing systems vs. unequal access). In each case, we would locate the value embedded in the STS on the continuum between the poles, identify the components of the STS in which it was embedded, and then look for misfits and possible value conflicts. While these concepts do not exhaust the value possibilities, they do provide a good start.

Neither this value mapping approach nor the STS x Issues grid are a better approach, but they each complement the other. Try them both.

A Problem Type Checklist

The table below presents a checklist that suggests ways in which ethical problems arise among actors or stakeholders in an STS. It will help uncover ethical problems embedded in the situation. The constituents of the checklist can overlap. For example, a problem could produce a harm (and violate the harm/beneficence test) and at the same time threaten the value of safety; in fact, threatening public safety could be the specific way it harms. But one can take advantage of this overlap to clarify the problem. The harm/beneficence test shows us that the general problem is preventing a harm; the concept checklist specifies this harm as a degradation of public safety. The checklist is a rough but helpful tool that suggests problems and provides categories to specify and clarify them.

Problem Type	**Definition**	**Checklist Questions**
Disagreements:	A *disagreement* is an ethical problem that arises when individuals hold different views of the facts or concepts present in a situation.	**Are there disagreements over facts?** **Are there any disagreements over concepts?**
Conflicts	*A conflict* occurs when two or more rights, values,	**Are there any conflicts?**

Problem Type	Definition	Checklist Questions
	goods, or interests appear mutually incompatible, that is, cannot both be maintained or promoted in a given situation.	**What are the conflicting values?** **(Moral vs. Moral? Moral vs. Non-Moral?)**
An ethical issue or concept becomes problematic	**privacy**: the privacy of individuals or groups could be violated due to a computing activity or computing system	**Are privacy rights in danger of being violated or superceded?** **What conflicting rights, goods, or values threaten to supercede privacy?**
	intellectual property: property rights could be threatened due to activities such as copying software	**Are intellectual property rights in danger of being violated or superceded?**
	safety: the safety of the public could be put at risk through the activities or products of computing professionals	**Could the safety of any constituency (especially the public be placed at risk?** **Which constituency?** **For what conflicting values is safety being traded off?**
	power: the increased power that accompanies computing (power created by knowledge that computing specialists have and by the way computing systems/products enhance human action) raises the issue of the responsible use of that power	**How do computers enhance or instrument human action in this situation?** **Does this enhancement lead to some harm such as deskilling, unemployment, or redistribution of power in social relations?**
An ethics test could be violated	Situations arise in which a computing specialist or computing product/activity could harm someone (violate the harm/beneficence test), treat someone with disrespect (violate the reversibility	**How could the reversibility test be violated?** **How could the harm/beneficence test be violated?**

Problem Type	Definition	Checklist Questions
	test), or threaten the integrity of individuals or groups (violate the publicity test).	**How could the publicity test be violated?**

Two issues require further discussion: disagreements and conflicts. In the following, we discuss the types of disagreements (factual and conceptual) and then conflicts (the types of which can be distinguished by the possible solution modes). Intermediate moral concepts (such as privacy, intellectual property, and responsibility) and the problems to which they give rise are discussed in the cases chapters (chapters 6 – 10).

Moral Disagreements:

A disagreement is an ethical problem that arises when the participants do not agree over some aspect of a situation. There are two types of disagreements: factual and conceptual.

Factual Disagreements:

A *factual* disagreement occurs when the participants do not know all the facts or have different versions of the facts. (Just what constitutes a fact is the subject of considerable philosophical debate. See refs) For example, the supervisor of company X orders a computer programmer to install a software program on a customer's (ABC Hardware) computers. The computer programmer believes that this would be illegal; the supervisor thinks it legal. Hence the supervisor and the programmer disagree on the legal facts of the situation: does copying the program for ABC violate the software license agreement?

One solves factual disagreements by uncovering more facts—in this case, the legal facts. The supervisor and the computer programmer need to consult the license agreement; they may also need to ask a lawyer for a legal interpretation of the agreement. But this disagreement can be readily solved; disputants often have ready access to the facts on which it hinges.

Sometimes factual disagreements are difficult to resolve. Here are some examples:

- *Disputed Historical Facts*: Parties may stand committed to different versions of past events. Maria Renato claims that Phil Port tried to kiss her after they had a business lunch last week. Phil denies this. Lacking independent evidence (a witness, a pattern of past behavior), this factual disagreement could prove intractable. (How should a manager respond to this problem given these different—even contradictory—claims about what happened?)
- *Technical and Scientific Limitations*: Has this software program been completely debugged? Current software testing methods fall well short of certainty. Do we adopt and use the software if *so far* we haven't found any bugs? (Use it unless it

is proven unsafe.) Do we delay use until we have conclusive evidence of safety? (Do not use it unless it is proven safe.) How much testing is enough? Can we reduce uncertainty to an acceptable level?

- *Difficulties in predicting future events* (especially long term future events): Risk assessment studies carried out for tanker traffic in Prince William Sound, Alaska predicted one major oil spill for every 242 years of pipeline operation, well beyond the expected lifetime of the Alaskan oil pipeline. Nevertheless, the Exxon Valdez oil spill, a major oil spill, occurred on March 29, 1989. Revised risk assessments revealed unacknowledged and hidden risks. So prediction of long term events, especially accidents involving complex technological, business, and social systems, comes up against limits in human foresight (as well as wishful thinking on the part of the oil companies sponsoring the studies). Risk assessments can be expressed only as probabilities, not certainties. Furthermore, even these probabilities must be set forth within a margin of error. Factual disagreements that rest on predictions and risk assessments are inherently difficult to resolve. (ref to Valdez and Cranor)
- *Limitations of time and money*: For example, a risk assessment study has validated that a nuclear storage site will contain radioactive material for 10,000 years. But as they neared completing their study (using a traditional risk assessment model), the engineers carrying out the study became aware of a more sensitive risk assessment model. Holding public safety as their paramount concern, they wanted to redo the study using the more advanced model. But the government agency funding the original study objected for two reasons. First, they were unwilling to spend more money on the study, and accused the engineers of trying to extend the contract for selfish reasons. Second, they pointed out that the temporary nuclear waste storage sites were inadequate and overused. Further delays could not be justified solely by the promise of uncovering more facts; they must be carefully weighed against the additional risk imposed by continuing to use the current inadequate, temporary storage sites. Perhaps the safety of the permanent storage site could be better established using the newer, more thorough risk assessment methods. But was this additional information worth the additional delay and expense? Limitations of time and money (and the additional risk of using current, temporary storage sites) argue against further investigation. Sometimes factual disagreements must be resolved on less than certain factual grounds. Sometimes the additional cost of continued study and the risks imposed by delaying action outweigh the supplementary benefits of accumulating new information. (There are, of course, additional factors to take into account in this case. How should the parties involved go about gaining the participation and consent of the public stakeholders in this situation? How do the risks and benefits stand to be distributed under the different alternatives of action?)

Not all factual disagreements can be eliminated, because not all the relevant or necessary facts are available. Historical, scientific, technical, time and cost limitations restrict our access to the relevant facts. As a result, there are times when action must be taken before all the facts are in. In this case, the uncertainty must be designed into the action. Engineers, for example, add a cushion to the safety factors they calculate to take into

account unforeseeable events. Actions can be taken that leave certain options open in the event that the future could offer more information. These decisions also have ethical implications. What do we do with uncertainty? How do we distribute the risks uncertainty imposes? The answers we give to these questions have profound ethical implications.

Conceptual Disagreements:

Sometimes, solving a factual disagreement doesn't produce consensus but, instead, reveals an underlying disagreement of a different nature. Suppose, for example, that a supervisor in company X and a computer programmer agree that copying software for ABC Hardware violates the software license agreement. Nevertheless, they continue to disagree on a course of action. The supervisor insists that the programmer copy the software. He argues that copying software is like driving 60 in a 55 mph zone: it breaks the law but does not harm anybody. On the other hand, the programmer argues that his professional responsibility requires that he respect property rights; he argues that rampant, unauthorized copying of software has a harmful collective and cumulative impact. The programmer rejects the analogy with speeding; for him it's more like shoplifting—something that is common but should not be tolerated. Each uses a different analogy to understand—and construct—the concept of ownership and the responsibilities it places on others. Their disagreement—at first factual—has now emerged at a different level as conceptual; they have constructed the concept of property—and the obligations it entails—differently.

This emergence of a conceptual disagreement changes both the nature of the disagreement and the method for resolving it. The disagreement is now deeper and more fundamental; the supervisor and computer programmer have substantially different ideas of software as property. Moreover, this new disagreement cannot be dissipated merely by uncovering more facts. They are faced with a more difficult project. To begin, each must understand what the other means by property. Then, they must work from their different concepts toward a common concept of property. If this is not possible, they should try to bracket the conceptual disagreement and reframe the problem, setting aside the conceptual issues about which they disagree and focusing on other issues on which they can agree. (For more on reframing problems and using different techniques—reasonableness and compromise—to solve them, see chapter 3.)

Resolving Conceptual Disagreements:

We solve conceptual disagreements by agreeing upon a definition of the concept in question. How do we do this? The following guidelines can help:

1. *Find a relevant rule, law, regulation, or guideline and use this to forge a common definition.* Your company uses catalyst A which is carcinogenic. Your supervisor claims that an exposure of 10ppm (parts per million) is safe while you argue that exposure should be no more than 2ppm. Furthermore, you both agree that exposure to 10ppm of catalyst A produces a 1 in 10,000 chance of contracting cancer. In short, you both agree on at least some of the relevant facts. Where you disagree is whether this risk is acceptable. Your supervisor argues that it is because it is comparable to other risks that the workers readily accept, like the

risk they take when they drive to work. You disagree because acceptable risk is determined (for you) by technical and economic feasibility. Reducing the exposure to 2ppm is technically feasible; it would involve the purchase of special machinery to filter out the catalyst. Moreover, even though this would cost the company some money, they can afford it. So you disagree with your supervisor and define the concept of safety in terms of the lowest level of risk that is technically and economically feasible. To resolve the disagreement, you and your supervisor could look to see if there are any legal standards. OSHA recommends an exposure of 5 ppm. You both then agree to modify your conceptions of acceptable risk to conform to the OSHA standard. This is useful because the procedure OSHA and other federal agencies use to set standards builds in the participation and consent of the public, i.e., those who will be placed at risk.

2. *Compromise.* In the above case, 6ppm is halfway between 2 and 10. You and your supervisor could resolving this conceptual disagreement by splitting the difference; lower the exposure to 6ppm because it is halfway between 10 (what your supervisor wants) and 2 (what you want). This works well in situations where the disagreement can be quantified, but accepting it too readily can lead to ignoring other possible solutions that you might generate if you negotiated more creatively.
3. *Reframe the problem.* Some conceptual disagreements prove intractable. Here it is advisable to change the problem. For example, you and your supervisor disagree over the meaning of safety. Perhaps, you could reframe this disagreement into the design problem of reducing the concentration of catalyst A in the working environment. Installing the latest, state-of-the art air filtering technology is one way doing this, but it may turn out to be expensive. Could non-technological (or different technological) procedures be designed to reduce the concentration of catalyst A with less expenditure? Perhaps it is not even necessary to reduce the concentration. *Worker exposure* could be reduced by developing more effective safety precautions (having workers wear masks), removing workers from the dangerous environment (developing a computer monitoring system), or reengineering the manufacturing process around a different, less dangerous catalyst. When you can't solve one problem, try formulating another.
4. *Look into how ethicists define the concept in question.* Several intermediate moral concepts such as safety, privacy, risk & reliability, and intellectual property have received extensive treatment in ethics literature. Examples can be found in this book, other texts on computer ethics (Johnson's for example), and in the many anthologies available (Huff & Fineholt, Johnson & Nussbaum, and Forrester). Disputants could consult this literature and adopt one of the definitions. This may appear to resolve the disagreement on the questionable grounds of authority, in this case the authority of a published ethicist. But ethical discussions involve spelling out and defending the conclusions reached. Examining, not just the definitions but the thinking process through which they are reached, makes this method more than just bowing to authority.
5. *A modified version of the casuistic method can help us reach common definitions.* Casuistry, a case-based method of analysis employed by medieval theologians,

provides a useful way to reach an agreement on concepts by first identifying model or prototypical instances of the concept and then comparing the questionable or borderline instances with the prototypes. You and your supervisor could identify instances that you both agree successfully embody the concept of safety (positive paradigms); you could also find instances that clearly do not embody safety (negative paradigms). Then you could classify the problematic case at hand by comparing it to the paradigms. For an example of this method, see the textbox below. This process takes time but is well worth the effort if it avoids a similar dispute in the future.

Text Box: Defining Intermediate Moral Concepts

A modified version of the casuistic method provides a useful way of defining the intermediate moral concepts that permeate the practice of computing. In this textbox we look at a classroom exercise carried out to define the concept of **corruption**.

1. *Identify prototypical instances of corruption*. Students looked at different cases posing ethical problems and sorted these out into those that raised corruption issues and those that did not; each developed a list without consulting with others. Then they refined the individual lists through group discussion. For example, they thought that an engineer who could only get a government construction contract by contributing to the campaign of the local mayor was dealing with corruption while an engineer who encountered opposition to his proposal to switch from a carcinogenic catalyst to benign one had a different kind of problem.
2. *Develop a preliminary features list*. After sorting out corruption from non-corruption cases, they set about identifying features common to corruption cases. A preliminary list cited stealing, injustice, illegality, and wrongdoing motivated by greed.
3. *Test the features*. The students tested each feature on different corruption cases. This helped eliminate irrelevant features, specify overly general features, and combine similar ones.
4. *Refine the features list.* The students produced a refined features list by working with additional cases, consulting essays on corruption, and interviewing different individuals (ethicists, business teachers, government officials). According to the refined list corruption involves…
    - **Secrecy**: The projects generated by corruption require secrecy in order to thrive.
    - **Irresponsibility**: Secrecy enables irresponsible action, that is, it allows people to act immorally while disavowing and disowning their actions and the resulting consequences.
    - **Diminution of individual integrity**: Secrecy corrupts individuals. Once they start to perform corrupt acts, their characters begin to disintegrate. The corrupt actions become easier when repeated and integrated into an individual's character. In this way, corruption diminishes an individual's integrity.
    - **Diminution of organization and institutional justice**: Corruption disrupts relations of justice within an organization. Rules are not fairly

applied. The benefits and harms associated with organizational activity are not fairly distributed. The result is an overall diminution of organizational integrity.

- **Conspiracies**: Corruption proceeds through groups composed of individuals united by solidarity with the project of corruption and the benefits it promises.
- **Greed**: Corruption is motivated by greed taken broadly to include not only desire for money but desire also for recognition, honor, and prestige.
- **Power**: Success in implementing a project of corruption depends on the power. Corrupt individuals illicitly use organizations, institutions, and technologies to magnify their power and achieve their goals.

5. *Use the refined features list to identify a positive paradigm, i.e., a prototypical instance of the concept.* Students reexamined several cases to identify positive paradigms of corruption, i.e., unproblematic examples of the concept. For example, they identified a paradigm case that involved an industry colluding to negotiate special tax deals with Puerto Rico's Internal Revenue Division. Members of this industry pressured employees of the PR IRS for special tax breaks not available to others.
6. *Identify a negative paradigm, i.e., a case that involves none of the features of the concept.* Students also reworked a positive paradigm case of corruption into a negative paradigm by removing all the corruption features. They rewrote the Town Z case summarized above to portray a fair bidding process instead of one that required secret campaign contributions to the mayor.
7. *Compare the problematic case with the positive and negative paradigms.* For example, students examined the following case:
    - *Marta Acevedo, an engineering student, has a laboratory exercise due tomorrow. She has been overwhelmed for the last few weeks with assignments from other classes and doesn't really have time to complete this exercise. She discovers that her roommate took this same class the previous semester and has the complete exercise on disk. She considers using her roommate's laboratory exercise*.

    Should Marta decide to use her roommate's lab exercise, then the success of her enterprise would certainly be facilitated by **secrecy**. Furthermore, should she cheat in this instance, it would be easier for her to do so the next time; so her course of action leads to **diminishing her moral integrity**. Moreover, it disrupts **relations of justice in the university** since students who have done the work are not treated fairly in relation to those who have not. However, her action is not enabled by institutional **power** nor is it motivated by **greed** (unless her final objective in getting her degree is the amount of money she will make when she gets a job). Moreover, it is not **conspiratorial**. So Marta's case falls mostly under the concept of corruption (since it involves secrecy and diminution of personal along with institutional integrity) but is not a prototypical case since it lacks other corruption features. On a continuum stretching from positive to negative paradigm, we would place her action closer to the corruption end of the scale than the non-corruption end. But we would have to acknowledge that a response to the problem in this scenario needs to be more complex than the

response to a prototypical corruption case.

Students concluded with a working definition of corruption. Corruption is a species of wrongdoing (i.e., it violates settled ethical principles or norms) that…

1. is facilitated by *secrecy* (under which one can act without accountability) and *power* (derived from one's position in some organizational hierarchy),
2. *diminishes the moral integrity* of those individuals involved by turning irresponsible action into habit,
3. *disrupts the justice relations* that obtain within and between organizations and institutions, and…
4. *is motivated by greed,* i.e., the desire to enrich oneself at the expense of others.

Conflicts:

Conflicts occur when two or more values appear mutually incompatible in a given situation. "Incompatible" means that it *appears* difficult to maintain, harmonize, or promote both values at the same time. For example, the computer programmer in the ABC Hardware case experiences a conflict between loyalty to her employer and adherence to professional ethical standards when her supervisor orders her to copy the program. She would like to satisfy both values but this appears difficult.

We respond to conflicts with three general strategies. Aim first *to design a creative middle way (win-win) solution that synthesizes, harmonizes, or balances the conflicting elements.* For example, the programmer could convince her supervisor that in the long run the company will benefit from pursuing ethical policies. Perhaps she could contact the software developer and get permission to copy the program for ABC Hardware for little or no charge. Or she could find another program that performs a comparable function and could be downloaded for free. All of these solutions allow the programmer to honor her obligation of loyalty to her employer while adhering to professional ethical standards. Always aim first to have your cake and eat it too.

Sometimes all efforts to synthesize the conflicting values fail. In these situations, we cannot, it seems, have our cake and eat it too. For example, suppose that the computer programmer tries to reason with her supervisor and persuade him to consider other options to copying the software illegally. However, the supervisor remains unmoved; he says there is no more time and questions her loyalty to him and the company. Then she must choose between loyalty and professional responsibility. After weighing her options, suppose she declines to carry out her supervisor's order; professional responsibility, she argues, has moral priority over loyalty. When we fail to design a creative middle way solution, we can fall back to the second best option: *look for the value that has moral priority and honor it while sacrificing the values that compete with it and are—morally speaking—less important.* A value trade off is not as good as a creative middle way solution. But it is the best option when a creative middle way solution is not feasible.

Finally, there are cases where conflicts cannot be resolved by finding a creative middle way solution or by honoring the conflicting element that has moral priority. We call these situations *dilemmas*. Here we face a forced choice between two or more mutually exclusive, morally equal options. It is literally a coin toss. As an example, many cite the decision facing Winston Churchill during World War II. British agents intercepted a coded message from the Germans announcing their intention to destroy a village in France. The agents decoded the message because they had earlier broken the German code. The Germans were unaware that their code had been broken, and the British wanted to keep it that way. This presented Churchill with a difficult choice: (1) save the village and tip off the Germans that their code had been broken, or (2) sacrifice the village in order to keep the Germans in the dark. If, indeed, Churchill was forced to choose between these two alternatives and both of them are morally equal, then Churchill faced a classic dilemma. (What do you think? Were these the only alternatives available? Are these alternatives morally equal?)

Moral imagination, which plays such a vital role in designing win-win solutions, is of no help in a dilemma. What is required, instead, is the determination to make the tough choice and see it through. The best strategy is prevention. Dilemmas most often arise when we fail to resolve moral disagreements or conflicts in a timely fashion. We ignore them in hopes that they will disappear of their own accord. But instead of disappearing, they get worse. For example, in the famous Ford Pinto case, engineers ignored safety considerations until late in the preproduction process, well after they had made commitments to a car design and had begun tooling the machines to stamp out the parts. According to one account, a Ford engineer scheduled a meeting on safety early in the design process and no one showed up. (ref to Dowie, *Pinto Madness*) When they finally turned to safety, they found a serious problem with the design and location of the gas tank. Because it was late in the preproduction process, and they had already made serious commitments to the current design, Ford chose not to respond to the problem. People died as a result. Moreover, losses due to damage suits and negative publicity cost the company billions, much more than they would have spent had they integrated safety upstream in the preproduction process or even at the late, downstream point where they first discovered the problem. There are two lessons to be learned from the Ford case. First, when confronted with a dilemma offering exclusive choices between safety and profits, choose safety. Second—and more fundamentally—to avoid hard trade off choices between safety and profit, integrate safety early and upstream in the preproduction, planning stage.

Often it is difficult to distinguish a disagreement from a conflict. The difference is this: conflicts are more severe than disagreements because they emerge out of a process of polarization where the difference between positions radicalizes into incompatibility. The conflicting positions are incompatible because they oppose one another; each claims that it alone is right (or good or virtuous) while the other is wrong (or evil or vicious). In opposition we frame the situation as a zero sum game where one side wins and the other loses. So conflicts emerge from disagreements when the latter, hardened by polarization, become closed off to reconciliation. Conflicts are nasty. They are more easily avoided

than resolved. Conventional wisdom holds here: an ounce of prevention is worth a pound of cure.

Writing the Report

Chapters 3 though 5 will take you through the process of designing and testing solutions to the problems that this chapter has allowed you to specify. Once you have done this, it is time to put all the pieces together into a report that you can use to educate your client about the social and ethical issues (and solutions) you have discovered.

Remember that your audience for the report is the client, not your instructor. You may find from your contacts with the client that only one person in the office will read it. Or you may learn that it will be widely shared. Your writing should focus on its appeal to the primary audience. But you need to be aware of the political issues that are always lurking when making recommendations to a client. Consultants are often brought into organizations to do an analysis and bring bad news of which the client is already aware. The role of the consultant can be a catalyst for the presentation of this news in an objective but sympathetic way that best allows for the news to be acted upon.

Even if you have not been set up in this way, any suggestion of change in a system can be taken by defensive members of your client's organization as implicit criticism. Your data collection should tell you what some of these sensitive areas are. Be thoughtful about how you make recommendations about them. Chapter 3 provides some guidelines to the organizational issues involved in what is called ethical dissent.

So you will need to be attuned to the political and practical issues of what can and cannot be done in your client's organization, how change occurs and at what pace, whether plain speaking or careful allusion is preferred, and what the differing agendas are of the different parties in the socio-technical system. And you will need to balance all these with the ethical responsibility you feel towards your client to give them the best advice possible.

The final report should consist of 6 sections:

1) an executive summary
2) a description of the system
3) the analysis of the results
4) recommendations
5) a reader's guide
6) a methodological appendix.

Executive summary. This should be a page or two summary of the report. It should include a description of the report and of the system, a discussion of the significant issues discovered, and a list of the top recommendations highlighted on the page. Each of these should be keyed to page numbers in the longer report. The idea is to provide a summary that a busy executive can read in 5 to 10 minutes to get the basic information about the report. Summarizing information in this way is in itself a useful skill for you to learn. Be aware that this summary is likely to be more widely shared than the full report.

Description of the system. This description should include the physical, logical, procedural, and social elements of the system. The physical structure includes the machines and other hardware involved, the networks, and the physical facilities in which the system is housed (e.g. the offices). The logical structure includes the data structures and software structures involved in the system. The procedural elements of the system include the ways in which data is gathered, collated, stored, backed up, and reported. They also include procedures for maintenance, repair, and replacement of the system, and any other relevant organizational procedures (e.g. those related to privacy protection or safety). The social elements of the system include a description of personnel and their relationships to each other and to other relevant stakeholders.

Analysis of the results. This section includes a discussion of those concrete aspects of the system that lead to specific concerns. These many include any single aspect of the system or interactions between aspects of the system (e.g. procedures that assume technical maintenance even though personnel are not trained). Patterns of use, patterns of oversight or error checking, specific hardware or software concerns, or specific organizational procedures are all candidates for inclusion in this section. The analysis of these specific concerns should highlight the specific risks associated with them, the probability of those risks occurring, and the likely harm or ethical concerns associated with those risks. Finally, the concrete advantages of resolving the specific concerns should be described. As you can see, this is analysis in the sense of an analytic presentation of issues rather than in the sense of statistical data analysis. Some statistics may be incorporated in this section, but the primary point is to present the social and ethical issues associated with the system.

Recommendations. This section should contain a set of recommendations or solutions that address each specific concern mentioned in the previous section. There should be at least two action options (preferably more) for each specific concern, and those options should be evaluated in terms of the client's goals, resources, and the ethical or social concerns involved. In most cases the client's goals will be multiple. For example, maintaining accurate records, guarding privacy, and minimizing cost. The effect of each option on this suite of goals should be noted. The options recommended should be carefully constructed to avoid simple black-and-white choices (e.g. safeguard privacy vs. disregard privacy) and to emphasize the best available options for dealing with the issue. Technical fixes (e.g. use a different backup method) should be included as options, but should not be the only options listed. For instance, procedural changes or personnel training could also be recommended.

Reader's guide. This should be a prose introduction to the most balanced and readable discussions of the issues that confront the client. It should include at least one item (e.g. a reader or an advanced article) that will serve as a window to further literature. This section can be organized as an annotated bibliography or as a prose review with references.

Methodological appendix. This section should contain a rationale for the particular methods chosen, and a detailed and concrete description of those methods. The individual interviews should be noted (though privacy issues here are important) as should the specific questions asked in the interviews and any changes made in the interviews to meet needs. The description of the field observation should include a description of, and a rationale for, the observation sites and times. It should also include a

description of the significant events looked for, a description of the significant events discovered, and a list of any changes made in the observation protocol made to meet needs. The description of the day-in-the-life scenarios should include a rationale for the choice of those particular perspectives and time frames, a description of the information from which they were compiled (e.g. interviews, manuals, etc.), and finally, the detailed scenarios themselves.