Technology Dialectics

A Paradigm for Cross-disciplinary Research and Development

by

Latanya Sweeney, Ph.D.

DRAFT
Fall 2008

How does an engineer or computer scientist make sure the technology she creates will be viable? How does she construct technology that is "provably appropriate" for a given personal, societal, organizational, and/or legal context? This writing introduces Technology Dialectics as a new research and development paradigm in which an engineer or computer scientist identifies and assesses potential barriers to technology approval and overcomes them through technology design. To accomplish this, an engineer or computer scientist harnesses cross-disciplinary knowledge. Scientific research methods (naturalistic observation, survey, interviews and experimentation) describe existing phenomena. Legal reasoning and policy interpretation situate phenomena in a socio-political framework. And, traditional computer science research involves methods for constructing new phenomena (that accomplishes a given task efficiently). Technology Dialectics blends different research traditions into a unified approach for developing technology such that the resulting technology is "provably appropriate" for a given context, and the basis of "proofs" are discipline specific. The approach of Technology Dialectics stems from lessons learned in constructing technologies that address real-world privacy problems. Examples come primarily from that venue, but this writing applies broadly to other application domains. Topics addressed within the Technology Dialectics paradigm include formulating problem statements, modeling real-world constraints, including end-user participation, determining validity, assessing generalizability, generating guarantees, and providing provably appropriate solutions.

Copyright © 2004-2008 Latanya Sweeney.

This book is dedicated

to my son

in the hopes he will always seek truth,

no matter where it leads.

Copyright © 2004-2008 Latanya Sweeney.

Acknowledgements

One never achieves an accomplishment alone but does so in environments and communities of people that help make it possible. This work is no exception. The Data Privacy Lab, the PhD Program in Computation, Organizations and Society, the students in the COS Lab course, the Center for the Quality of Life Technology, and family and friends all played supporting roles.

Much gratitude goes to the students, faculty, staff and corporate partners of the Laboratory for International Data Privacy ("the Data Privacy Lab")[1] for providing an environment in which privacy technology was explored, regardless of discipline – policy, law, technology – and with the dogma of delivering provably appropriate solutions to real-world problems. The rigor students demanded of themselves and the vision we all shared fostered a transdisciplinary environment that provided fodder for this book, as the examples within give witness. In alphabetical order, specific gratitude goes to: Edoardo Airoldi, Sylvia Barrett, Guy Blelloch, Lenore Blum, Manuel Blum, Samuel Edoho-Eket, Aarthi Gopal, William Gronim, Ralph Gross, Ponnurangam Kumaraguru, Yiheng Li, Sherice Livingston, Kishore Madhava, Bradley Malin, William Malloy, Elaine Newton, Raj Reddy, Rebekah Siegel, Michael Shamos, Brooke Singer, Ben Vernot, and Marshall Warfield.

Tremendous gratitude goes to my colleagues in the PhD Program in Computation, Organizations and Society at Carnegie Mellon University's School of Computer Science for their unwavering confidence in my pursuing this endeavor as part of the research training of their students. Appreciation goes to Kathleen Carley, Norman Sadeh, William Scherlis, Mary Shaw, Raj Reddy, Michael Shamos, and Lorrie Cranor.

Many thanks go to the students in the PhD Program in Computation, Organizations and Society who worked through and commented on earlier versions of this writing. In alphabetical order, thanks to Virginia Bedford, Michael Benisch, Eric Daimler, George Davis, Serge Egelman, Ian Fette, Il-Chul Moon, and Michael Schneider.

Appreciation goes to the NSF Center for Quality of Life Technologies for extending this work beyond privacy technology into robotics, rehabilitation engineering, and general engineering. Particular appreciation goes to Kate Seelman, Rory Cooper, Takeo Kanade, and Jim Osborn.

Last but not least, here is a word about family and personal advisors. From the conception of this work, my family whole-heartedly believed in this effort and sacrificed personal time for its completion. The earliest seeds for this work stem from conversations with Henry Leitner at Harvard University and the late Herb Simon at Carnegie Mellon University many years ago. Thank you all.

Multiple groups contributed to funding this work. While all contributions were important, two deserve particular recognition. The Intel Corporation made a donation through Lab Partnership 50526. Special thanks to David Hoffman, Director of Privacy and Security Policy, and Claire Vishik, Trust/Security Standards and Regulations Manager. The National Science Foundation gave support, in part, through grant EEC-0540865.

[1] More information on the Data Privacy Lab is available at privacy.cs.cmu.edu.

Copyright © 2004-2008 Latanya Sweeney.

Any opinions, findings, conclusions or recommendations expressed in this book are those of the author and do not necessarily reflect the views of any of the people or groups listed in this section.

Copyright © 2004-2008 Latanya Sweeney.

Chapter 1 The Need for Contextually Appropriate Technology

This book introduces Technology Dialectics as a way for an engineer or computer scientist to construct technology, utilizing knowledge across disciplinary boundaries as needed, such that the resulting technology is provably appropriate[2] for a given personal, societal, organizational, and/or legal context. The goal is for new technology to have a better likelihood of user acceptance, societal adoption and organizational uptake.

This chapter examines:

- how inter-, multi- and trans- disciplinary research effect disciplines.
- computer science's disciplinary origin and its relationship to other disciplines.
- reasons for a computer scientist or engineer to expand her research approach to include other disciplines.

1.1 A goal is to conduct cross-disciplinary research

While computer science is a single discipline, there are increasing pressures for a computer scientist to solve problems that not only require the inclusion of multiple disciplines, but that also demand the integration of research methods across disciplines. Working with different disciplines is at the heart of interdisciplinary and multidisciplinary work, which computer scientists and engineers do readily. Integrating research methods from other disciplines into computer science itself is something else altogether. Yet, as computer scientists construct technology for increasingly complex contexts that involve different kinds of stakeholders, the need grows for computer scientists to consider other perspectives during design.

A computer scientist or engineer often develops technology or conducts research in a vacuum, talking only to her like-minded colleagues and ignoring available knowledge from other disciplines such as anthropology, economics, law, or psychology. Often the vision of a single computer science community guides development decisions and sets research directions for that community in the absence of external, relevant knowledge, even knowledge from other computer science communities. A single project often reflects the vision of a single practitioner. Reasons for these unilateral pursuits include past success and the overhead of working across community boundaries.

Historically, many projects launched from isolated computer science perspectives have been hugely successful. After all, the sole vision of computer scientists led to many of today's computing accomplishments. Examples include popular personal computer programs such as electronic mail, spreadsheets, word processing, and web browsers.

[2] The term "provably appropriate" is a strong standard introduced in upcoming chapters. Given a technology, a context in which the technology operates, and operational constraints imposed on the technology by the context, the technology is provably appropriate for the context if it satisfies the operational constraints. The basis of a "proof" is discipline specific.

Copyright © 2004-2008 Latanya Sweeney.

Increasingly however, using only a monocular computer science perspective does not work. An example is the serial number in the Intel Pentium III processor.[3]

> **Example 1.** In January 1999, Intel announced that Pentium III processors[4] would include a serial number in order to help combat the growing number of computer break-ins occurring over the Internet [1]. Intel's vision was to enable stronger Internet security by embedding within each computer an unalterable unique identifier. When a computer with a Pentium III processor engaged in communication over the Internet, the communication could include the processor's unique identifier, thereby associating the computer's actions to its physical processor. Sales and usage records could further relate the processor to an owner and location.
>
> Within two months of the announcement, privacy and consumer groups filed complaints with the Federal Trade Commission [2]. Their complaints argued that an identifying serial number could also allow unknown observers to track computer use across the Internet, even if the user was not being malicious or doing anything wrong. Supplemental complaints explained that surreptitiously tracking the movements of individuals on the Internet violated expectations of anonymity and of fairness and control over personal information [3].
>
> By April 2000, Intel acquiesced and attested that future processors would not have a serial number. Computer break-ins over the Internet continue to be a serious problem.

Abrupt abandonment of research after lengthy and expensive investment is costly to developers and to society. Rather than risk abandonment at the end of a development process, some researchers and investors want assurances that developed technology will have a maximum likelihood of acceptance and adoption. This book describes a way to conduct research that integrates knowledge and methods from multiple disciplines, as appropriate, in order to help guarantee user acceptance, social adoption, and organizational uptake of resulting technology.

1.2 Interdisciplinary, Multidisciplinary, Transdisciplinary

When discussing research across disciplines, the terms *interdisciplinary*, *multidisciplinary* and *transdisciplinary* describe the effect of the research on the disciplines. Colloquially, the terms interdisciplinary and multidisciplinary are interchangeable and refer to any research effort involving more than one discipline. But understanding their semantic differences is important. Interdisciplinary research is a means to fuse disciplines. Multi-disciplinary research is a means to fuse results from different disciplines. And transdisiciplinary research fuses knowledge and methods regardless of discipline.

A discipline is a corpus of knowledge and a system of problem-solving methods, techniques, and skills. Examples of disciplines include anthropology, biochemistry, computer science, linguistics, mathematics, physics, psychology, and zoology.

[3]Another example is the abandonment of research projects on ubiquitous technologies due to privacy concerns. See the discussion on the Total Information Awareness Program in Section 1.6.

[4] A computer's processor interprets instructions and processes data contained in computer programs. It is the "brains" of the machine.

Copyright © 2004-2008 Latanya Sweeney.

The life of a discipline is dynamic. Research often creates specialized fields of studies that evolve into disciplines. Once a discipline solves all of its key problems or its problems are no longer valued, it could then self-eradicate. The knowledge and know-how of disciplines do not die. Instead, newer or broader disciplines absorb them. Interdisciplinary and multidisciplinary research play important roles in the creation and development of disciplines.

The terms interdisciplinary, multidisciplinary, and transdisciplinary describe different ways of engaging disciplines in jointly solving problems. The means of identifying, formulating, and solving a shared problem differ based on how disciplines engage.

In interdisciplinary research, researchers from two or more disciplines synthesize their disciplinary approaches to better address the problem. Typically, the resulting approach leads to changes in problem-solving in the originating disciplines and/or introduces a new field. In fact, interdisciplinary research is fundamental to the creation of most new fields.

There are numerous examples. Fusing statistics and machine learning brought forth computational data mining. Bioinformatics combines molecular biology with computer science. Quantum information processing blends quantum physics and computer science. Other disciplines that began as interdisciplinary research include biochemistry and biomedical engineering.

Computer science started as an interdisciplinary effort that fused electrical engineering and mathematics. In the early 1970's, virtually no professor in a computer science degree program had a degree in computer science. Most often, these professors held degrees in mathematics. Over the last 30 years, computer science emerged as a discipline and then joined other disciplines in interdisciplinary research to establish even more fields.

In multidisciplinary research, researchers from two or more disciplines work on a common problem, but do so without altering their disciplinary approaches or developing shared methodology. Think tanks offer good examples of multidisciplinary research. Industry and the military often fund multidisciplinary research as a means to solve narrowly defined problems quickly.

Conducting multidisciplinary research usually involves sub-dividing the problem along disciplinary lines. Typically, multidisciplinary research begins by partitioning the shared problem into discipline-specific sub-problems. Researchers within disciplines then work somewhat alone on their sub-problems. Discussions with researchers in other disciplines are often limited to interface and requirements issues. The final solution to the overall problem merges discipline-specific results.

In comparison, researchers conducting interdisciplinary research tend to focus on all aspects of the same problem jointly, even though they do so from different approaches initially. It is not surprising that interdisciplinary research tends to require interactive, on-going communication throughout the research process.

Transdisciplinary research[5] uses multiple disciplines to help researchers better understand a research problem. This approach is advantageous when a single discipline lacks the ability to understand all the dimensions of the problem. There is no emphasis on disciplinary fusion or discipline-specific solutions, because there is no emphasis on disciplines at all. The emphasis is

[5] Jean Piaget first coined the term "transdisciplinarity" in 1969 as being between, across and beyond disciplines.

Copyright © 2004-2008 Latanya Sweeney.

on using different disciplinary approaches and knowledge to gain insight into the problem. As a result, transdisciplinary research can remove disciplinary impasses that may block problem understanding and solving.

Interdisciplinary and multidisciplinary research are specializations of transdisciplinary research, but transdisciplinary research is more than their aggregate. It offers its own form of knowledge production. Unlike interdisciplinary and multidisciplinary research, which focus on disciplinary integration (of methods and results, respectively) within disciplinary frameworks, transdisciplinary research focuses on the integration of multiple perspectives regardless of discipline.

The "context" in which the technology operates is becoming increasingly complex and specialized. In earlier years, a computer scientist could often focus on general purpose technology whose context was limited to a compliant user. Today, many research problems intimately weave technology into intricate human, organizational or societal contexts. Transdisciplinary research is useful because it provides a means to understand context that is often multi-faceted and complex and defies disciplinary terms.

Nowotny et al. examined recent conflicts between science and society and found that context often involved many stakeholders for which no dialogue existed [4]. Therefore, resulting science provided little or no understanding of their perspectives.

In recent examples where developers sought to include context throughout the design and development process, more involvement on the part of society led not only to better social solutions, but to better technical results as well. Hughes, an emeritus professor who studies sociology and technology, gave as examples the Massachusetts Institute of Technology's SAGE air defense project, the Atlas intercontinental ballistic missile project; Boston's Central Artery/Tunnel project, and the Internet [5].

> **Example 2.** The development of the Defense Department's Arpanet project, which led to the development of the Internet, provides an example of the importance of context. Much of the Internet's innovativeness stems from fusing the social and technical contexts of the scientists who created it. Hughes claims that the decentralized and anarchical nature of the Internet reflects the alienation of many members of the scientific community from the military over the Vietnam War [5]. These scientists were no longer willing to gather in military-based working groups as scientists had previously (and since). These scientists' favor for autonomous and non-centralized technology drove design, and their values show in the result. In this case, the scientists did not have to study the attitudes of the end-users because they were the end-users. In general, research involving stakeholders beyond the scientists themselves requires methods for infusing stakeholder input into the design process.

So much emphasis on working across disciplines makes it easy to forget that disciplinary pursuits are important too. Many intellectual endeavors and the source of many intellectually challenging problems arise largely within a single discipline. Disciplinary work often examines broader and more generic classes of problems than the problems tackled in interdisciplinary, multidisciplinary, or transdisciplinary work. Disciplinary work sheds light on today's problems, and on problems that may become of interest in the future.

Copyright © 2004-2008 Latanya Sweeney.

Transdisciplinary and disciplinary research complement each other. Disciplinary know-how feeds transdisciplinary research; and in exchange, transdisciplinary research fertilizes disciplinary knowledge.

This book introduces a transdisciplinary research approach to a new researcher so her resulting technology will benefit from diverse perspectives throughout technology design. In presenting its paradigm, this book introduces research methods and approaches across disciplines, and is therefore a good primer for researchers engaged in multidisciplinary or interdisciplinary research.

In this book, the term *cross-disciplinary* research is used to refer to inter-, multi-, and trans- disciplinary research without distinction. Cross-disciplinary research refers to research involving multiple disciplines regardless of how the disciplines engage. Because interdisciplinary and multidisciplinary research are special cases of transdisciplinary research, using "cross-disciplinary" is the same as using transdisciplinary generally. Of course, transdisciplinary research is more than the sum of interdisciplinary and multidisciplinary, and many references in this book to transdisciplinary research refer to activities that are neither interdisciplinary nor multidisciplinary. The distinction of using "transdisciplinary" generally or "transdisciplinary" as excluding interdisciplinary and multidisciplinary should be obvious from the written setting.

1.3 Computer science has always engaged different disciplines in isolation

One historical perspective of computer science research advocates using engineering and mathematical research methods to construct technology, and using scientific research methods[6] to understand the technology in context. Historically, a computer scientist or engineer used engineering and mathematical methods to construct technology as a solution to a problem. In turn, the technology, when used in context, generated contextual phenomena studied using scientific research methods. Differences between engineering/mathematical research methods and scientific research methods are huge. They include ways of describing problems, standards for validating results and techniques for generalizing knowledge. While historically computer science has engaged scientific research methods, it has generally done so by relegating scientific methods to the study of post-construction problems. The approach taken in this book incorporates scientific and other research methods throughout technology design and evaluation.

In 1967, three pioneers of computer science at Carnegie Mellon, Alan Newell, Alan Perlis, and Herb Simon, wrote an article in Science entitled, "What is Computer Science?" in which they defined computer science as "the study of the phenomena surrounding computers" [6].

Their definition, as expounded in their essay, recognized: (1) computer science research as the construction of complex instruments – an engineering pursuit; (2) computer science research as the study of algorithms – something of a mathematical pursuit; and, (3) computer science research as understanding the natural phenomena surrounding the existence of the

[6] "Scientific research methods" consist of naturalistic observation, surveys, interviews, case studies, and experimentation. Forthcoming chapters introduce the details of scientific research methods, as well as other research methods. Computer science heavily relies on engineering and mathematical research methods, which are not the same.

Copyright © 2004-2008 Latanya Sweeney.

machine in its environment – a scientific pursuit. These correspond to three different kinds of research methods –engineering, mathematics, and scientific methods, respectively.

While much of computer science research historically relied on engineering and on algorithmic methods for conducting research, recent decades gave rise to additionally studying the computer environment. Examples of these newer studies include topics like inequitable Internet access ("the digital divide") [7], the social interactions of people using the Internet [8], and privacy tradeoffs on the Internet [9]. These studies use scientific research methods, which are not engineering or algorithmic methods.

In summary, different research methods have been involved in computer science research, but pre- and post- construction boundaries have tended to dictate applicability. Pre-construction has heavily relied on engineering and mathematical methods, and post-construction has heavily relied on scientific methods. This book introduces ways to incorporate many more research methods, not just scientific methods, into pre-construction to better control post-construction phenomena.

1.4 Today's computational systems need integrated research methods

There is no doubt that traditional approaches to computer science research have revolutionized the world in which we live, but past success using computer science research methods is not a guarantee for adequately facing new challenges. As computers permeate personal, business and social life, computer science research itself must expand to explore the development of computational technology appropriate for these environments. Doing so requires incorporating other kinds of research methods into technology design in order to identify and characterize potential acceptance, adoption, and uptake barriers.

Understanding the environments in which computational solutions will operate is not easy for a technology developer. It involves working with different terminology, different research methods, and different ways of interpreting, validating and generalizing results. Here are some examples from privacy of the difficulties faced.

> **Example 3.** In 2003, Carnegie Mellon hosted the Privacy in Data workshop which brought together some of the world's leading computer science theorists to examine real-world data privacy problems [10]. The biggest hurdle (consuming the most time) involved helping brilliant computer scientists understand the personal, organizational and policy settings in which real-world data privacy problems exist. To understand the language and tenets of other disciplines takes time. While everyone spoke English, discipline-specific vocabulary, standards and principles limited communication with lawyers, policy experts, and domain specialists.

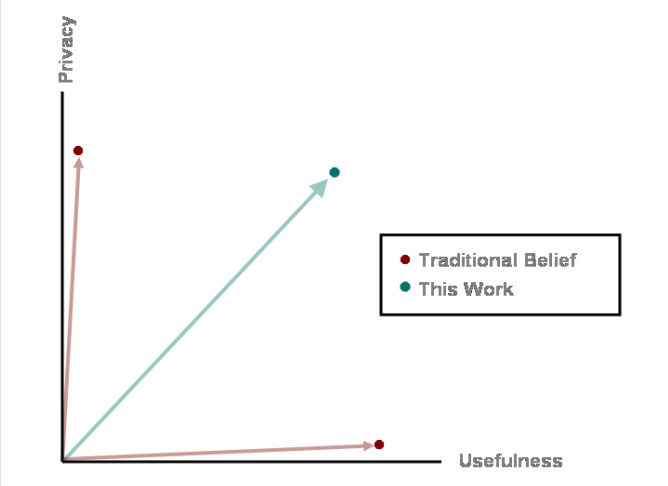
Copyright © 2004-2008 Latanya Sweeney.

Example 4. On the other hand, ignoring the real-world context is not desirable either. Privacy-preserving data mining has become a major theme at leading ACM, IEEE, and AAAI[7] conferences. Many works in privacy-preserving data mining tend to result from a traditional monocular computer science pursuit without sufficient attention to real-world details. As a result, many published solutions are mathematically sound and impressive, but naïve in terms of real-world social, economic, and policy settings [11]. Published results are therefore of limited (or no) practical utility because they are a mismatch to real-world privacy problems.

Example 5. Another approach of limited use involves inviting experts to "fix" a privacy-invasive technology after the prototype exists. That's typically too late in the process because significant opportunities to remove barriers within the technology are lost. The technology already exists. Eligible remedies are limited to adding patchwork policy and post-production packaging.

Example 6. Consider examples of technologies that offer safety through invasive surveillance (discussed in more detail in Section 1.6). After 9/11, safety has been a priority among Americans. Yet, early surveillance technologies gave little or no attention to privacy during their design. Afterwards, society faced the choice of either adopting the technology (as is) and losing privacy, or having privacy without the promised safety. See diagram below.

When faced with a choice, American society chose privacy [12]. But did society need to choose at all? Could society enjoy both privacy and safety? Better choices are often available when those who develop technologies address privacy problems during design [13][14]. But how does a developer do that?



[7] Abbreviations for the leading computer science professional organizations: Association for Computing Machinery (ACM), Institute of Electrical and Electronics Engineers (IEEE), American Association for Artificial Intelligence (AAAI).

Copyright © 2004-2008 Latanya Sweeney.

This book offers a paradigm for constructing technology that is accountable to its post-production context. Upcoming chapters explain many details, but here is a nutshell description. A computer scientist or engineer assumes responsibility for including context within her design by fusing research traditions from multiple venues, as needed, in order to learn design constraints imposed by context. Guarantees that the resulting technology satisfies contextual constraints form the basis for asserting the technology is "provably appropriate" for its context.

1.5 A goal is to assume responsibility for the viability of resulting technologies

In the old days, computer scientists tended to be of the opinion that the goal of computer scientists was to construct machines and the goal of social scientists was to get society to accept them. As discussed earlier, computer science methods have historically been concerned with engineering and mathematical perspectives, whereas social science methods have described phenomena related to the technology's context after construction. This is not a formula for the future. Instead, a technology developer must herself fuse methods and knowledge across disciplines into the technology she builds. Doing so ensures viability for her technology.

Why does the responsibility of technology communing with its context reside with a computer scientist or engineer? It doesn't solely rely on her, but she does construct tomorrow's machines, and in so doing, she can overcome many barriers to acceptance and adoption within her design. When a computer scientist or engineer understands contextual constraints prior to designing technology, the technology design may overcome these barriers. This is a unique advantage that a computer scientist or engineer has over other social scientists and other parties. She can remove some potential barriers without users, organizations or society ever seeing them.

More generally, there exists a class of user acceptance, societal adoption, and organizational uptake problems for which the best solution lies within technology design. This book helps a technology creator identify and solve these problems within her original design.

In the absence of computer scientists and engineers removing adoption barriers, these problems become ones for policymakers, social scientists, economists, business, and information scientists to solve. While each of these have their own methods capable of producing their own brand of solution, none of them have the unique opportunity afforded the originating computer scientist and engineer.

Consider what happens when a technology faces societal adoption barriers. Below are perspectives of how policymakers, public policy researchers, and information scientists may consider remedies to address the conflict.

New technology often finds itself in uncertain gray areas when developers give no consideration to policy during design. Technology that requires legislative or regulatory change for adoption faces serious troubles. Policy response may be insufficient and unpredictable. Here are some reasons why. First, policymakers work in a framework of principles and practices that govern human and organizational behavior, which is often a mismatch to technological evolution. An answer to a technological challenge today can have unforeseen consequences for tomorrow's technology. Second, policymakers enact policy in elastic time –at times dragging and at other times fast paced. Rarely is the process steady and deliberate, leaving the fate of the

Copyright © 2004-2008 Latanya Sweeney.

technology uncertain for an unspecified time period. Third, policy language is limited to words which crudely describe allow/prohibit provisions. Crude descriptions may fail to capture the subtlety of the technological issue. This book introduces computer scientists and engineers to ways to think about policy during design to avoid legislative, regulatory or judicial uncertainty.

> **Example 7.** As an example of a legislative mismatch, consider the video capture of people in public. American jurisprudence[8] long since determined that anyone may photograph a person in a public space without his or her consent. In the 1970's many states enacted wiretapping laws that prohibit recording phone calls, even if the person doing the recording is a party to the conversation. All parties must consent. Consider camcorders, which combine continuous photography with sound recording. Solely as a consequence of the breadth of the wording of some state wiretapping laws, video recording a person's activities (walking, eating, reading, swimming, sleeping, etc.) in public is possible without consent provided the recording does not reveal the person's conversations [15]. Unfortunately, virtually all camcorders, available in retail stores today, record with sound and provide no means of recording without sound. The popular acceptance of camcorders predicts upcoming legal challenges for some camcorder uses.

Public policy researchers are often social scientists who use economic or statistical science to help remove barriers to technology adoption by assessing historical data. Public policy studies tend to provide helpful, descriptive insight into the nature and character of adoption barriers. There are two shortcomings. First, knowing about a barrier is not the same as removing it, and understanding a conflict is not the same as resolving it. While policy research can inform regulatory change and technology development, the outcome of policy research is rarely a solution unto itself. Policymakers and/or technology developers must use policy research results to remove barriers or resolve conflicts. Second, realizing the benefits of public policy research post technology production is a lengthy process. It takes years. This book recommends ways to reap the benefits of social science research early in the technology design process.

> **Example 8.** As an example of public policy research's ability to encourage changes in policy and technology, consider recent work on the economics of consumer privacy decisions. A historical cornerstone of consumer privacy practice is the notion of consent. (An applicant consents to sharing her credit report with a credit card company.) Consent assumes that the individual makes sharing decisions with knowledge of possible consequences. Micro-economists recently provided evidence that consumers often lack enough information to make these kinds of consent decisions in today's setting [16]. When faced with benefits afforded by personalization technologies (e.g. loyalty cards at stores and email lists), consumers provide personal information readily. It is almost impossible for a consumer to understand the long-term privacy trade-off in the face of seemingly unrelated short-term benefits. One problem is secondary data sharing [17]. Even if a consumer scrutinizes the original data sharing agreement, the flow of personal information beyond the originating party is unseen and typically does not require consent. Other parties may receive consumer information, and it may eventually flow into potentially harmful insurance, employment, and financial decisions [18]. These findings underscore the need for critical re-assessments of personal data policies and related technologies.

When facing a technology adoption barrier, information science (IS) and information technology (IT) researchers often use existing technical know-how to invent add-ons. IS/IT results tend to be external modifications. When such a remedy is available and appropriate, IS and IT

[8] American jurisprudence refers to the sum of all laws in the United States. This includes the U.S. Constitution, federal and state statutes, federal executive orders, and court-made law.

Copyright © 2004-2008 Latanya Sweeney.

researchers solve the problem. So, there is a class of adoption problems best solved by IS/IT researchers. Since the technical know-how exists beforehand, then in many of these cases, the originating computer scientist or engineer could solve the problem earlier, during the design process. This book provides strategies for the original developer to learn about obstacles to acceptance and adoption early in the design process.

> **Example 9.** As an example of an IS/IT solution to a pressing problem, consider credit card purchases over the Internet. In 1994, almost as soon as web browsers became available, retail merchants sought to sell items using the Internet. The problem was how to do so without allowing an eavesdropper to learn credit card information. Early attempts were awkward and clumsy. Some merchants required shoppers to download and use special programs before they could transmit their credit card numbers [19]. Other merchants required part of the transaction by telephone. While computer scientists had strong encryption methods, a lack of standards for incorporating encryption technology into web browsers was the biggest problem. Finally, in 1994 IS/IT developers at Netscape introduced the Secure Sockets Layer Protocol (SSL) as a cryptographic means for securely facilitating data transfers over the Internet [20]. In 1996, Netscape announced the first web browser that included a powerful encryption program for users to send encrypted messages to similarly-equipped Web servers using SSL. The process was seamless. SSL became the standard immediately and remains the fundamental means of conducting securing transactions over the Web. In 2006, U.S. shoppers spent $211 billion online using SSL [21].

Clearly, policymakers, public policy researchers, and information scientists provide many worthy contributions beyond removing technology adoption barriers. Additionally, there are problems for which they offer the best solutions. This discussion narrowly describes by contrast, how the a computer scientist or engineer can do more to construct technology that avoids foreseeable adoption barriers by infusing work from policymakers, public policy researchers, and others, early into her design process.

Be warned. Having a computer scientist or engineer remove potential barriers to adoption during the design process can save time and resources, and can offer a broader spectrum of remedies, but it is a benefit not a panacea. It is not feasible to believe that a computer scientist or engineer can remove all possible barriers herself, even with the early help of social scientists, lawyers, and others. Laws may still need to change. Some barriers may escape early detection. Post production issues and barriers may still arise. While there is a class of problems for which computer scientists and engineers can predict adoption barriers and remove them, some problems will remain.

Copyright © 2004-2008 Latanya Sweeney.

1.6 Example: Computer science research and privacy

Examining privacy issues in computer science research provides motivating examples for producing technology that is provably appropriate for its context. A computer scientist or engineer wants to ensure the viability of the technology she constructs. Ignoring privacy concerns until after construction (or design) has often faced grave consequences. Rather than ignoring privacy, a technology developer must learn to construct privacy safeguards into the technology she builds in order to ensure its viability.

Previous sections in this chapter underscored a growing need for developers to include contextual issues in design. This section offers an illustration by summarizing recent events in computer science research related to privacy. The following sub-sections include situations where research funding was lost, almost lost, or may be lost over privacy concerns.

Example 10. Computer science research and privacy

In 2003, Congress abruptly ended research on the Total Information Awareness (TIA) project funded by the Defense Advanced Research Projects Agency (DARPA) due to privacy outcry. The suddenness left many computer scientists without funds, notwithstanding scientific results and financial commitments to students. Many researchers felt vulnerable and helpless. Shortly afterwards, a dialogue among computer scientists was held at Carnegie Mellon University to discuss growing privacy concerns in computer science research [22]. Below is a synopsis of related events and discussion.

Alarmist reactions to privacy-invasive technologies

Recent news articles ignited public concern over issues of privacy in emerging technologies. But when presented in alarmist ways, public and legislative response can be so drastic as to inhibit scientific research and progress and thwart possible benefits to society. Perhaps the technology and research are changing so fast that the consequences are more than the public, reporters, lawmakers, or even computer scientists can realize or prepare for. Here are some reactions in the news.

> "The Defense Advanced Research Projects Agency (DARPA) stimulates outside-the-box thinking that has given us the Internet and the stealth bomber. On occasion, however, DARPA goes off half-cocked. Its Total (now Terrorist) Information Awareness (TIA) plan to combine all commercial credit data and individual bank and academic records with F.B.I. and C.I.A. dossiers, which would have made every American's life an open book …." [William Safire, "Dear Darpa Diary," New York Times, June 5, 2003]

These kinds of characterizations of the TIA project by Safire and other reporters generated serious privacy concerns eventually terminating the project and nearly ending related computer science research which had broader personal and societal benefits.

In January 2003, Senator Feingold, Democrat-Wisconsin, introduced legislation to place a moratorium on data mining research and deployment efforts at the U.S. Department of Defense. Senator Wyden, Democrat-Oregon, introduced a similar anti-data mining bill, but it was limited to TIA. A broad coalition of public interest groups, ranging from the American Civil Liberties Union to the American Conservative Union urged Congress to take action against TIA. [23]

Copyright © 2004-2008 Latanya Sweeney.

Congress did respond, but in doing so, data mining research barely escaped what could have been a horrible blow to funding.

Privacy concerns are not limited to TIA, but are inherent in many emerging technologies.

Two years earlier, in January 2001, police in Tampa, Florida tested face recognition technology during Super Bowl XXXV, scanning faces of people in crowds, comparing them with images in a database of digital mug shots. In August of that year, a councilwoman in Jacksonville, Florida introduced legislation banning the use of face recognition technology by the Sheriff's Office and other city agencies. [24]

Other city councils and legislatures considered similar legislation, but the events of September 11, 2001, dramatically reversed the projected down-turn expected for funding in face recognition research.

These recent encounters are viewed as warnings by some computer science researchers, who worry that future encounters may lead to serious loss in research support and who themselves have concerns over the perception and use of the technology they create.

Technologies that record and/or share person-specific information are at the center of these privacy concerns. The growth of data capture technologies is due to the widespread availability of ever smaller machines having faster processors with larger, less expensive storage capacities, communicating over ubiquitous networks. As the cost of collecting data approaches zero, more technologies that capture and share data are likely.

Personal promise of data intensive technologies

Privacy issues appear heightened when the benefactor of the data collection is not the subject of the data. In the earlier examples, society at large benefited from the individual's loss of privacy. What happens when the benefactor is an individual?

Personal assistant technologies are examples of emerging technologies that require a lot of personal data. Unlike the surveillance examples in the previous subsection, these technologies are typically for the direct benefit of at least one of the subjects of the data even though privacy concerns remain. Here are some examples of uses of personal assistant technologies.

Modern computer users spend a large amount of time searching for desired information and organizing the information they currently maintain on a desktop. Executives in companies hire secretaries to manage much of this information. Computers capable of managing personal communication may dramatically improve productivity (and possibly reduce stress) by helping schedule and process the bombardment of phone calls, faxes, and email messages that typifies the contemporary office.

Ubiquitous computing over networks of sensors and cameras with coordinated data collection and sharing may yield personal benefits to aging populations and those living with debilitating diseases. Many elderly patients have to give up independent

Copyright © 2004-2008 Latanya Sweeney.

living because they forget. They forget to visit the restroom, to take medicine, to drink, or to see the doctor. Personal assistants may help. Using cameras and microphones in a patient's home, professional care-givers can interact directly with remote patients. Systematic monitoring may allow patients to avoid a range of emergency conditions.

Computer scientist attitudes about privacy

Reactions by computer science researchers to privacy issues in their research are crudely: (1) "technology trumps privacy;" (2) "technology research is policy neutral;" and, (3) "computer scientists assume responsibility." Here is a brief summary.

In the "technology trumps privacy" position, computer science researchers take stock in past accomplishments and computational benefits enjoyed by society, thereby relying on a belief that if society must choose, it will choose advancements in computer technology over privacy [25].

Some go further and recommend avoiding any attention to privacy concerns until after society enjoys the benefits of the technology. Warnings against this position caution that unforeseen dangers could be unleashed forever.

In the "technology research is policy neutral" position, computer science researchers do not contemplate any privacy or social implications that may be inherent in the construction or existence of the technology they seek to build. Instead, these computer scientists want to pursue their research, leaving any related privacy issues to social scientists, policy makers, lawyers, and others. Warnings have been voiced that such positions are themselves human value decisions, and computer science researchers cannot escape making them [26].

In the "computer scientists assume responsibility" position, computer scientists want to be active in recommending and enabling remedies. Some believe assuming responsibility is a necessary condition to insure viability of future technology.

Today, many computer scientists must insulate their creations from risks related to user acceptance, societal adoption and organizational uptake. The approach taken in this book provides a way for a computer scientist or engineer to work across disciplines, as needed, to identify and address possible barriers early in the development process.

1.7 Funding sources want effective research involving multiple disciplines

While this book uses research across disciplinary boundaries as a means to ensure acceptance and adoption of developed technology, funding sources are driving researchers towards inter- and multi- disciplinary research primarily to expand science. Because researchers go where the money is, researchers are increasingly writing grants for research across disciplines. However, there are concerns about how to conduct such research, especially if more than two disciplines are involved. This book poses an approach.

Funding sources encourage research to involve multiple disciplines because they believe doing so is the only way to solve some kinds of problems, especially problems that traditional disciplines seem unable or unwilling to address. In fact, many scholars believe that the most

Copyright © 2004-2008 Latanya Sweeney.

pressing problems of our times, including the AIDS pandemic, global warming, and privacy, can be solved only by developing approaches across disciplinary boundaries.

In 2004, the National Academy of Sciences hosted panel discussions with federal, private, and international funding organizations, researchers, research center directors and educators and held a public comment session regarding interdisciplinary research. Findings recommended federal agencies adjust their policies to help improve research by encouraging collaborative and interdisciplinary work [27].

In January 2005, a memo from the White House Office of Science and Technology Policy announced it would allow research projects to have more than one principal investigator in an attempt to drive multidisciplinary science [28].

The National Science Foundation, U.S. Department of Energy, the U.S. Department of Defense, and the National Institutes of Health provide substantial grants for computer science research. Each of these has special allocations for research involving multiple disciplines.

Funds provided to research that engages multiple disciplines are growing. It is difficult to report the exact research dollars awarded by the National Science Foundation for projects involving multiple disciplines because some disciplines lie within the same broad scientific categories. Nevertheless, funding grew for projects that involved disciplines so sufficiently broad from one another as to lie outside different scientific categories (from $270 million in constant 1987 dollars in FY 1990 to $860 million in constant 1987 dollars in FY 1996) [29].

This is happening internationally too. The Multidisciplinary Research Program is a multi-agency program of the Canadian Department of Defense that supports research teams whose efforts intersect more than one traditional science and engineering discipline [30]. Typical award funding levels range from half a million to a million U.S. dollars per year for three to five years. U.S. universities are eligible for these awards.

While funding agencies push research that engages multiple disciplines, few people have figured out how to make it work in general.

Of course, successful interdisciplinary research has sparked new fields and led to multidisciplinary ones as well. For example, the fusion of design and computer science launched the field of human computer interaction. Today, human computer interaction draws on expertise not only from computer science and design, but also from psychology, sociology, and anthropology [31].

Despite inter- and multi- disciplinary successes, difficulties remain. How does a researcher or engineer generally leverage other disciplines to help in individual research? This book provides a framework for the individual researcher to assess and conduct research across disciplinary boundaries.

Copyright © 2004-2008 Latanya Sweeney.

1.8 Technology Dialectics offers a unified approach

This book introduces "Technology Dialectics" as a paradigm for conducting cross-disciplinary research and development. The goal of Technology Dialectics is to help researchers and developers produce technology that is provably appropriate for a given context. Doing so ensures acceptance, adoption, and uptake of resulting technology.

This chapter expressed a need for computer scientists and engineers to construct contextually appropriate technology. Previous subsections described this as a timely pursuit that requires working across multiple disciplines. Subsequent chapters will describe how a computer scientist or engineer accomplishes these feats. This section ends this chapter with a general notion of the thesis presented in remaining chapters.

Technology Dialectics is a new research paradigm in which researchers evaluate and understand the requirements and implications of the technology they seek to construct throughout design and development. To accomplish this, researchers and engineers enhance their problem statements such that technical solutions to these expanded problems resolve potential clashes before they become barriers to adoption. The overall goal is to identify and resolve operational conflicts early in the design process in order to ensure user acceptance, societal adoption, and organizational uptake of the end result.

Technology Dialectics fuses methods across disciplines as needed. Scientific research methods (naturalistic observation, survey, interviews and experimentation) describe existing phenomena, whereas traditional computer science research involves methods for constructing new phenomena (that accomplishes a given task efficiently). Technology Dialectics blends these different research traditions into a unified approach for developing technology that is provably appropriate for a given societal, organizational, policy, and/or legal context.

Technology Dialectics is broader than recent efforts. Participatory design [32], for example, describes ways to include those who will use the technology ("end-users") into the technology design process. Value-drive design [33] represents stakeholder interest as equations and then resolves conflicting interests based on an optimally solving the system of equations. In comparison, Technology Dialectics describes ways to incorporate design constraints important to lots of different kinds of stakeholders, including society and organizations, and does so using a broad set of methods and disciplinary knowledge that are not limited to expression as equations and analysis that may force tradeoffs.

The approach of Technology Dialectics stems from lessons learned from constructing technology that address real-world privacy problems, as examples throughout this book give witness. However, Technology Dialectics is broader than privacy technology. Other application areas include e-business, technology policy, technology for the aging, and national security.

Topics addressed within the Technology Dialectics paradigm include formulating problem statements, modeling real-world constraints, including end-user participation, determining validity, assessing generalizability, generating guarantees, and providing provably appropriate solutions. Subsequent chapters present Technology Dialectics in detail.

Copyright © 2004-2008 Latanya Sweeney.

Here is a summary of this chapter. Inter-, multi- and trans- disciplinary research effect the formation of disciplines and describe ways of using disciplinary knowledge in problem-solving. Computer science provides an example of a discipline created through interdisciplinary pursuit, and once established, active in fostering other new disciplines. However, the time seems ripe for computer scientists and engineers to rethink disciplinary approaches to problem-solving methods. Ensuring user acceptance, societal adoption, and organizational uptake are reasons to expand research approaches to include other research methods in problem-solving.

The next chapter identifies personal barriers to cross-disciplinary research that can limit success. Subsequent chapters introduce the mechanics of research methods and the Technology Dialectics paradigm.

Copyright © 2004-2008 Latanya Sweeney.