Representation, Inclusion, and Innovation

Multidisciplinary Explorations

Synthesis Lectures on Human-Centered Informatics

Editor

John M. Carroll, *Penn State University*

Human-Centered Informatics (HCI) is the intersection of the cultural, the social, the cognitive, and the aesthetic with computing and information technology. It encompasses a huge range of issues, theories, technologies, designs, tools, environments, and human experiences in knowledge work, recreation and leisure activity, teaching and learning, and the potpourri of everyday life. The series publishes state-of-the-art syntheses, case studies, and tutorials in key areas. It shares the focus of leading international conferences in HCI.

Representation, Inclusion, and Innovation: Multidisciplinary Explorations
Clayton Lewis
November 2017

Research in the Wild
Yvonne Rogers and Paul Marshall
April 2017

Designing for Gesture and Tangible Interaction
Mary Lou Maher and Lina Lee
March 2017

From Tool to Partner: The Evolution of Human-Computer Interaction
Jonathan Grudin
December 2016

Qualitative HCI Research: Going behind the Scenes
Ann Blandford, Dominic Furniss, and Stephann Makri
April 2016

Learner-Centered Design of Computing Education: Research on Computing for Everyone
Mark Guzdial
November 2015

The Envisionment and Discovery Collaboratory (EDC): Explorations in Human-Centered Informatics with Tabletop Computing Environments
Ernesto G. Arias, Hal Eden, and Gerhard Fischer
October 2015

Humanistic HCI
Jeffrey Bardzell and Shaowen Bardzell
September 2015

The Paradigm Shift to Multimodality in Contemporary Computer Interfaces
Sharon Oviatt and Philip R. Cohen
April 2015

Multitasking in the Digital Age
Gloria Mark
April 2015

The Design of Implicit Interactions
Wendy Ju
March 2015

Core-Task Design: A Practice-Theory Approach to Human Factors
Leena Norros, Paula Savioja, and Hanna Koskinen
March 2015

An Anthropology of Services: Toward a Practice Approach to Designing Services
Jeanette Blomberg and Chuck Darrah
February 2015

Proxemic Interactions: From Theory to Practice
Nicolai Marquardt and Saul Greenberg
February 2015

Contextual Design: Evolved
Karen Holtzblatt and Hugh Beyer
October 2014

Constructing Knowledge Art: An Experiential Perspective on Crafting Participatory Representations
Al Selvin and Simon Buckingham Shum
October 2014

Spaces of Interaction, Places for Experience
David Benyon
September 2014

Mobile Interactions in Context: A Designerly Way Toward Digital Ecology
Jesper Kjeldskov
July 2014

Working Together Apart: Collaboration over the Internet
Judith S. Olson and Gary M. Olson
November 2013

Surface Computing and Collaborative Analysis Work
Judith Brown, Jeff Wilson, Stevenson Gossage, Chris Hack, and Robert Biddle
August 2013

How We Cope with Digital Technology
Phil Turner
July 2013

Translating Euclid: Designing a Human-Centered Mathematics
Gerry Stahl
April 2013

Adaptive Interaction: A Utility Maximisation Approach to Understanding Human Interaction with Technology
Stephen J. Payne and Andrew Howes
March 2013

Making Claims: Knowledge Design, Capture, and Sharing in HCI
D. Scott McCrickard
June 2012

HCI Theory: Classical, Modern, and Contemporary
Yvonne Rogers
May 2012

Activity Theory in HCI: Fundamentals and Reflections
Victor Kaptelinin and Bonnie Nardi
April 2012

Conceptual Models: Core to Good Design
Jeff Johnson and Austin Henderson
November 2011

Geographical Design: Spatial Cognition and Geographical Information Science
Stephen C. Hirtle
March 2011

User-Centered Agile Methods
Hugh Beyer
2010

Experience-Centered Design: Designers, Users, and Communities in Dialogue
Peter Wright and John McCarthy
2010

Experience Design: Technology for All the Right Reasons
Marc Hassenzahl
2010

Designing and Evaluating Usable Technology in Industrial Research: Three Case Studies
Clare-Marie Karat and John Karat
2010

Interacting with Information
Ann Blandford and Simon Attfield
2010

Designing for User Engagement: Aesthetic and Attractive User Interfaces
Alistair Sutcliffe
2009

Context-Aware Mobile Computing: Affordances of Space, Social Awareness, and Social Influence
Geri Gay
2009

Studies of Work and the Workplace in HCI: Concepts and Techniques
Graham Button and Wes Sharrock
2009

Semiotic Engineering Methods for Scientific Research in HCI
Clarisse Sieckenius de Souza and Carla Faria Leitão
2009

Common Ground in Electronically Mediated Conversation

Andrew Monk
2008

Copyright © 2018 by Morgan and Claypool

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopy, recording, or any other except for brief quotations in printed reviews, without the prior permission of the publisher.

Representation, Inclusion, and Innovation: Multidisciplinary Explorations
Clayton Lewis
www.morganclaypool.com

ISBN: 9781681732480 print
ISBN: 9781681732497 ebook
ISBN: 9781681732503 hardcover

DOI 10.2200/S00812ED1V01Y201710HCI038

A Publication in the Morgan and Claypool Publishers series
SYNTHESIS LECTURES ON HUMAN-CENTERED INFORMATICS, #38
Series Editors: John M. Carroll, Penn State University

Series ISSN: 1946-7680 Print 1946-7699 Electronic

Representation, Inclusion, and Innovation

Multidisciplinary Explorations

Clayton Lewis
University of Colorado, Boulder

SYNTHESIS LECTURES ON HUMAN-CENTERED INFORMATICS #38

MORGAN & CLAYPOOL PUBLISHERS

ABSTRACT

A representation is a thing that can be interpreted as providing information about something: a map, or a graph, for example. This book is about the expanding world of computational representations, representations that use the power of computation to provide information in new forms, and in new ways. Unlike printed maps or graphs, computational representations can be dynamic, and even interactive, so that what is represented, and how, can be shaped by user actions. Exploring these new possibilities can be guided by an emerging theory of representation, that clarifies what characteristics representations must have to express the meaning being represented, and to enable users to discern that meaning easily and accurately. The theory also shows the way to inclusive design, for example using sounds to represent information commonly presented visually, so that people who cannot see can understand what is being presented. Because representations must be shaped by the abilities of their users, and by the nature of the meanings they convey, creating them requires perspectives from multiple disciplines, including psychology, as well as computer science, and the sciences appropriate to the content being expressed. The book presents a series of explorations of this large and complicated space, as invitations to further study, and to innovation.

KEYWORDS

representations, inclusive design, visualization, interactive simulations, aesthetics, visual programming

Contents

Preface

> *What characterizes … new media are their unprecedented dynamics, based on their underlying computational mechanisms. …[W]e need the creative elaboration of the particular dynamic capabilities that these new media afford and of the ways that through them humans and machines together can perform interesting new effects. These are avenues that have just begun to be explored, primarily in the fields of new media, graphics and animation, art and design. Not only do these experiments promise innovations in our thinking about machines, but they also open up the equally exciting prospect of new conceptualizations of what it means to be human, understood not as a bounded, rational entity, but as an unfolding, shifting biography of culturally specific experience and relations, inflected for each of us in uniquely particular ways.*
>
> –Suchman (2007), p. 23.

Representations are all around us: a map represents a place, a picture can represent a person, a curve drawn on paper can represent a mathematical function, this book represents a collection of ideas. We'll get more technical about just what representations are, and how they work, but for now we'll use examples like these to suggest that a representation is a thing that can be interpreted as providing information *about* something, that is, about whatever is being represented.

The human world has always been full of representations, as far back as we can see, but it's never been fuller than now. That's because of *computation*, whose power is highlighted by Lucy Suchman in the quotation above. The reason computation is so useful, in almost everything that people do, is because of the fabulous resources it offers for new ways of representing things, all kinds of things. These new representations have miraculous advantages over what's been available until now. At the dawn of writing a representation of a few words would be a lump of clay, of substantial size and weight; today, all the books ever published could be fit into a storage device that would fit in your hand; see http://www.gizmodo.co.uk/2016/12/every-book-ever-published-would-fit-on-to-one-hard-disk/. Further, computational representations of books can be sent to the other side of the world with virtually no delay, they can be copied with essentially no delay, and at essentially no cost, and so on.

And we aren't at all limited to representing things like the contents of books. We can create dynamic, animated presentations; what it took the Disney studios weeks to do not long ago can be done today by a schoolchild at their desk. We can create sounds, music, and synthetic speech, all computationally, meaning that these representations share the virtues of lightness, smallness, ease of transmission, and ease of replication of computationally represented text. But more than that, we can control and vary the production of any of these things very easily. A program that creates music won't produce just some particular piece of music, it will let you create the music you want (if you have the patience and skill). And you can interact with the program. It doesn't just produce whatever representation it makes, on its own; it shows you what it is doing and invites you to modify it, while production is going on. An atlas was once a collection of printed maps. If you wanted a different map you had to get a different atlas. Now you can use computational representations that allow you to shape what you want the map to show, and how, with enormous flexibility, and change the view you get as you get it, so as to focus on what you want to see, even if you didn't know what you wanted to see when you started.

Box I.1

The sociologist Bruno Latour, in his essay, "Visualization and Cognition: Thinking with Eyes and Hands" (1986), has described the profound impact of *inscriptions*, things like maps, drawings, diagrams, and tables, on the progress and practice of science. He lists nine properties of such representations that make them especially useful.

They can be *moved* easily from one place to another.

They *don't change* when they are moved.

They are *flat*, and hence content is not hidden.

Their *scale is flexible*, so that structures of very different sizes can be rendered conveniently.

They can be *reproduced* cheaply.

They can be *recombined*, so that information about different things can be brought together. One form of recombination is superimposition.

They can be *made part of a written text*.

They can be analyzed using *two-dimensional geometry*, so that (for example) the sizes of things can be readily measured.

We can see that computational representations intensify some of these benefits; for example, computational representations can be moved and reproduced even more easily than inscriptions on paper, such as Latour studied. Operations like rescaling, recombination, or measurement also become easier and cheaper. But the potential of computational representations goes beyond intensifying the benefits of inscriptions to providing new ones: interactive control, moving images, the integration of sound and image, and much more.

One of the consequences of the enormous new potential of computational representations is to greatly expand the audience for information of all kinds. It used to be that blind people could only read newspapers if someone else read the material and recorded it, or transcribed it as Braille. Today, almost all newspapers provide computational representations of their content that can be presented to anyone as synthetic speech. This is an example of the potential of computational representations to support *inclusive design*, that is, the design of representations that can be processed effectively by the widest audience.

This book is about this expanding world of computational representations, including its potential for inclusive design. I'm writing it because, exciting as what we can now do every day is, there's more that we can do. I also think that the frontiers of what we can do, the new possibilities, are *interesting*, as well as important, and I want to share the excitement I feel about them.

So this book isn't meant to be a report of what I or other people have already done. Rather, it's a report of *explorations*, attempts to push back those frontiers. As you'll see, the explorations are all incomplete, and some are, so far at any rate, failures. Maybe none of them will prove to have mapped territory that people in the future will find especially useful. But I hope to show something of what's out there, and what you will see if you go there yourself. I'm hoping that you'll find more than I have.

When exploring something, it's helpful to have some kind of map, however rough, within which to sketch the new things you find. I've found it helpful to use a theory for this, a *theory of representation*, that is broad enough to encompass the wide range of content and technology with which we will be concerned. As discussed in the next chapter, this theory is built on a substantial mathematical foundation, something called the *representational theory of measurement*. It adds vital insights from the work of Jock Mackinlay, who showed how this kind of theory can expose the key challenges in creating any representation, of anything: *expressiveness*, that is, the ability for a representation to be faithful to the structure of what's being represented, and *effectiveness*, the requirement that the operations that have to be performed to use a representation, for example comparing the lengths of two bars in a bar chart, can be performed easily and accurately by whoever has to perform them. One can see that what's effective for one person won't necessarily be effective for another: someone who can't see the bars can't readily compare their lengths. This is the aspect of the theory that supports inclusive design.

As we've seen, representations today can be highly dynamic, and interactive. I sketch how the theory can be expanded to provide a useful way to think about these kinds of representations, as well as the familiar static ones that sit still on a page or on a screen.

This theory requires ideas from more than one discipline to deploy it, which is to say that ideas from more than one discipline are required to develop new representations. The theory distinguishes two domains, a *target* domain, which is whatever we are actually interested in, for example world affairs, and a *representation* domain, which contains whatever the representation itself is, for example words printed on paper, or synthetic speech. Building new representations requires ideas about both domains. For the representation domain, we may need to know about printing, or (for all the explorations reported here) computer programs, the displays they produce, and how to support possible ways of interacting with them. We also need ideas from the target domain: what is it about world affairs that we care about? What does the representation need to help us do?

The effectiveness requirement brings in even more ideas, quite different ones. If our representations are going to be consumed by people, we need to know what people can do easily, and what they can't. This is psychology. I hope you will enjoy having all these ideas, from different fields, rattling around together.

As we develop this theory we'll see how it relates to ideas that are familiar to many computer scientists and programmers, the separation of presentation from content, and design patterns that separate *models* from *views*. These ideas take advantage of a natural *layering* that appears in many representations: the representation domain for one representation becomes the target domain for another. So a system can contain a model, that represents some content of interest, but cannot directly be observed by a user; a view then represents the model in a way that the user can observe. To be useful, the view has to be effective, in Mackinlay's sense, and the composite of model and view must be expressive, that is, it must faithfully reflect the structure of the underlying content.

All the explorations in the book involve computational representations, and nearly all the chapters include computer programs that illustrate the ideas. The programs can be accessed from http://claytonhalllewis.github.io/bookPage.html. These programs are not finished work, so please don't expect them to be. I am a shocking programmer, and not at all a "software engineer;" that is, the programs are full of things no one should do. I shouldn't have done them, either, but in trying out the ideas, as they developed, I just didn't make the investment I should have to make the programs clear and clean. The programs have hardly been tested at all; as the wise Antranig Basman has suggested, such programs are "not even software." So please don't take any of the programs as exemplary as programs.

One of the many wonderful things about the world we now live in is that programs like these can be provided to you in a form that you can not only play with, but also modify, to try new ideas of your own, or to make improvements. All of them are written in Javascript, and should run for you in Chrome, Firefox, or Safari, with no need for you to install anything on your computer. Most use no code that isn't already in your browser; for the few exceptions I've included copies of the other code in what's available to you. There's nothing exotic. If you are new to Javascript, or for that matter to programming, there are excellent materials online, for example https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics, if you know how to create a web page, or https://www.w3schools.com/js/ if you do not. The latter site lets you play with the language in your browser without setting up a web page of your own.

The target domains for the representations in the explorations are varied. The explorations in Chapters 2–4 are about physics concepts. The material is drawn from a large collection of interactive simulations for learning physics developed by my colleagues at the University of Colorado, Boulder, and made available online at https://phet.colorado.edu. They key interest here is, how can one make interactive simulations like these work for people who don't see well? The target domain for Chapters 5 and 6 are programs, written in "visual languages" intended to make programming easier for beginners. Here again the challenge is, how can the conceptual benefits of these "visual languages" work for people who can't work with visual material? A popular programming activity for beginners is turtle graphics, a simple way to make a program draw pictures. Chapter 7 explores how turtle graphics might be made to use sounds rather than drawings. Many people feel that music conveys spatial movement to them as they listen. Chapter 8 explores whether this could be developed as a way to convey graphical information in sound. Chapter 9 explores how forms in more than three spatial dimensions might be represented. Such forms are easy to represent mathematically, but most people find that representation difficult to understand. Can we develop representations that are easier to understand? Finally, the target domain for the exploration in Chapter 10 is programs, again, but here the kind of programs that people in the arts sometimes use. Might it be possible to develop representations of programs that have more aesthetic potential than the textual and visual languages that we have today, and that might have other advantages, such as being more directly intelligible? I've not been able to do this, but perhaps you can.

Many people have helped with this work, by contributing ideas and suggestions, or as collaborators on some of the projects. Of course none of them is to blame for the roughness, or downright wrongness, of the ideas and the programs. They include Tamer Amin, Sina Bahram, Antranig Basman, Alan Blackwell, Beat Brogle, Bill Casson, Hunter Ewen, Michael Eisenberg, Noah Finkelstein, Inge Hinterwaldner, Varsha Koushik, Richard Ladner, Owen Lewis, Emily Moore, Steve Pollock, Alex Repenning, Derek Riemer, Ben Shapiro, Taliesin Smith, Andreas Stefik, and Jason White. Frieder Nake kindly provided examples of his work. Prof. Mehul Bhatt of the University of Bremen welcomed me into his research group. The ATLAS Institute at the University of Colorado Boulder supported some of the work. Colin Clark and an anonymous reviewer read the manuscript in draft and made many valuable suggestions.

For the programs that accompany the book I am indebted to the wonderful ecosystem of freely-available code to do all kinds of things, that the web offers us. Of special value are Blockly, a library for creating blocks languages, created by a group led by Neil Fraser of Google; Flocking, a sound processing system created by Colin Clark of OCAD University; and Raphaël, a library that makes SVG graphics much easier, provided by Dmitry Baranovskiy. The Stack Overflow community, both the people who ask questions, and those who answer them, provided enormously helpful information.

I owe more than I can express to the Hanse-Wissenschaftskolleg, in Delmenhorst, Germany, its Rector, Prof. Dr. Reto Weiler, and its wonderful staff, including Dr. Dorothe Poggel, Research Manager for the Brain area at the HWK. A six-month fellowship in residence at the HWK provided an opportunity for reading, programming, and discussion that made it possible to undertake the book. The international community of fellows, staff, and associates, including Tamer Amin, Margarita Balmaceda, Ann Blake, Jacopo Dal Corso, Marion Daniel, Susanne Fuchs, Alessa Geiger, Christina Gehrking, Petra Heinz, Stefan Heinz, Kim Hoke, Ian McDonald, Heidi Mueller-Henicz, Lucy Pao, Claire Raymond, AJ Reese, Brandi Reese, Thierry Ribault, Susanne Schregel, Amritashis Sengupta, Dipa Sengupta, Nicole Schuck, Elizabeth Sheffield, Li Shu, Wolfgang Stenzel, and Ilka Weniger provided an environment both intellectually stimulating and socially enjoyable. I thank all concerned with this wonderful institute, including the community members in the Verein der Freunde und Förderer des Hanse-Wissenschaftskollegs in Delmenhorst e.V. who do so much to support it. I also thank my colleague Gerhard Fischer, now emeritus, for decades of collaboration, and for telling me about the HWK.

Finally, I thank David Krantz for including the theory of measurement in the Experimental Psychology Proseminar at the University of Michigan in the Fall of 1973.

CHAPTER 1

Theory of Representation

Informally, a representation is a thing that provides information about something else, the way a map provides information about a place. But although this definition provided a starting point for our discussion in the last chapter, we need to move beyond it. A huge problem with it is that it says that a representation is a thing, but in fact a thing, in itself, can't be a representation.

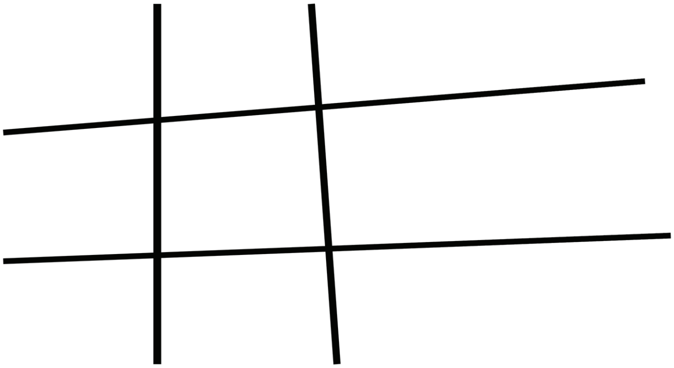

Figure 1.1: Lines in a grid, forming a tic-tac-toe board, or a map, or … .

How can this be? Aren't maps things, and aren't maps representations? Figure 1.1 brings out the trouble. Is that diagram a rough map of a portion of downtown Boulder, Colorado? Or is it a tic-tac-toe board? Or is it a rough map of a portion of lower Manhattan? As a thing, in itself, the diagram could be any of these things, or none of them… there is no way to know. Only when accompanied by some interpretation, that specifies what the diagram is a representation of, does it become a representation.

Box 1.1

Ruth Millikan and What Things Represent

The philosopher Ruth Millikan argues that things can be representations by virtue of their causal history: something can be considered a representation if the way it came into being depended on its representing something (Millikan, 1989). If when I drew the diagram in Figure 1.1 I drew what I did because I wanted a map of Boulder, then it is a map of Boulder.

More interestingly, a structure in an organism can be said to be a representation if its contribution to the evolution of the organism depends on its representational role. For example, Millikan argues that a particular structure that is sensitive to magnetic fields, in certain tiny marine organisms, actually represents the direction of oxygen-poor water. In the part of the ocean where these creatures live the earth's magnetic field aligns with gravity, which is difficult for tiny organisms to sense, "Down" is the direction of oxygen-poor water. The direction of the magnetic field, in itself, isn't important to these organisms; neither is gravity. But the direction of oxygen-poor water is important. So the structure is there because of the information it provides about oxygen-poor water, and, Millikan says, that's therefore what it represents.

Relating Millikan's thinking to our framing, can a thing in itself be a representation, for Millikan? Perhaps so: she might say that that structure represents something, and what it represents is determined by its history, whether we know the history or not. Thus, Figure 1.1 is or isn't a map of Boulder, whether we know it or not: its history, if we knew it, would answer the question.

Suppose I am discussing Manhattan, and I reach for Figure 1.1 to aid in my description. Then, in that setting, the figure would be a representation of Manhattan, because that's why I introduced it. This would be true, in that setting, even if whoever originally drew Figure 1.1 intended it to be a map of Boulder.

We'll sidestep these perplexities by not considering that the figure is a representation except when considered along with an interpretation. We won't ask whether or when the needed interpretation could be recovered from the figure's history. Rather, we'll consider that a thing like Figure 1.1 is a representation only when considered as a part of a *representational system* that provides the interpretation we need.

The representational theory of measurement (RTM; Krantz et al., 2007) provides a starting point for developing these ideas. RTM describes measurement as establishing a *structure-preserving mapping* between an empirical relational structure and a mathematical relational structure. We'll refer to these structures as the *target domain*, a domain in which we have practical interest, and the *representation domain*. The "structure preserving mapping" is the interpretation we need to tell us that something is being used as a representation, of what, and how.

Measurement of length provides a familiar example that illustrates these ideas. Suppose we are in the situation depicted in Figure 1.2, in which we need to bridge a chasm, and we have a supply of logs. We want to know whether a given log will bridge the chasm. We could do this by actual trial, but that would involve hard work, and possibly even danger, as we maneuver the log at the brink of the chasm.

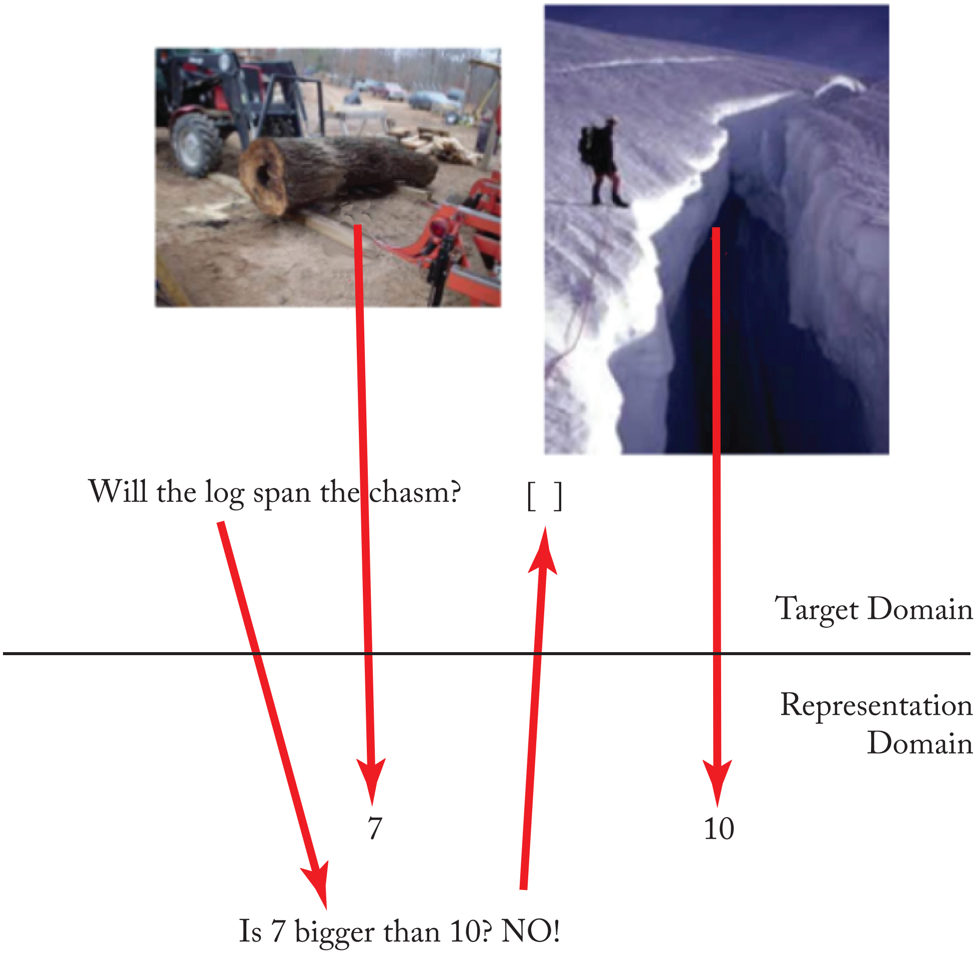

Figure 1.2: Using measurement to determine if a log will span a chasm.

Of course we realize that we can use measurement to answer this question, without moving the logs at all. We measure a log, and we measure the chasm (using a little trigonometry, if necessary), and we compare the numbers we get. If the number we get from measuring the log is bigger than the measurement of the chasm, the log is long enough; otherwise, it's not.

Now suppose none of the logs is long enough, but we have some means of joining logs end to end. Now we want to know if some combination of logs will be long enough. How can we tell how long a combination will be? Here again, we could do the physical work of connecting the logs, and then measure them. But we can avoid this work using measurement: we know that we can predict the length we can span by connecting logs end to end by *adding* the lengths of the separate logs.

These facts about measurement are an everyday miracle. How can it be that we can predict the results of physical actions, like laying a log across a chasm, or connecting logs together, by manipulating numbers? See Box 1.2 for discussion.

Box 1.2

Why does measurement work?

The physicist Eugene Wigner, in a famous essay, "The Unreasonable Effectiveness of Mathematics in the Natural Sciences" (1995; first published 1960), points out the surprisingness of the correspondence between physical phenomena and relatively simple mathematical structures. He offers no explanation of it: "The miracle of the appropriateness of the language of mathematics for the formulation of the laws of physics is a wonderful gift which we neither understand nor deserve."

Wigner cites the pragmatist philosopher Charles Sanders Peirce on this:

"The great utility and indispensableness of the conceptions of time, space, and force, even to the lowest intelligence, are such as to suggest that they are the results of natural selection."

...

"Such an hypothesis naturally suggests itself but it must be admitted that it does not seem sufficient to account for the extraordinary accuracy with which these conceptions apply to the phenomena of Nature, and it is probable that there is some secret here which remains to be discovered." [Peirce, 1878].

Another commentator is the mathematician Richard Hamming (1980):

"I have tried, with little success, to get some of my friends to understand my amazement that the abstraction of integers for counting is both possible and useful. Is it not remarkable that 6 sheep plus 7 sheep make 13 sheep; that 6 stones plus 7 stones make 13 stones? Is it not a miracle that the universe is so constructed that such a simple abstraction as a number is possible? To me this is one of the strongest examples of the unreasonable effectiveness of mathematics. Indeed, I find it both strange and unexplainable."

Hamming suggests that the apparent miracle owes something to selection: we pay more attention to situations that are well described by mathematics than to those that are not, and most things are not. But he concludes that the fit, where it happens, still needs an explanation that he can't provide.

A related question is, why are we so completely confident of the predictions that we make by mathematical means? In Hamming's example, why are we so certain that combining a flock of six sheep and a flock of seven sheep will always give a flock of 13 sheep? If we were to try the experiment, and get a different result, we would be sure that we had made a mistake somewhere. Why?

The pioneering developmental psychologist Jean Piaget argued that certain mathematical structures become fundamental to people's reasoning as they mature, changing status from possibilities to inevitable certainties (Piaget, 1971; Piaget and Voyat, 1979). But later work has shown that these certainties can be undermined by contrary evidence. For example, Hall and Kingsley (1968) used covert removal of clay to "show" that rolling a ball of clay into a thin plate made it weigh less, violating the principle of conservation of substance, one of Piaget's structures. Many college students accepted the faked results. Later studies suggest that the details of the inquiries matter, but all show some tendency for adults to accept anomalous results (see, e.g., Winer et al., 1992). So it does not appear that certainty in these matters is securely established during development.

We don't get very much actual data, in the course of everyday life, about the effectiveness of measurement, yet we accept sweeping generalizations about it. For example, we've likely never measured two lengths, one of more than a mile, and one less than a foot, and verified that we can predict their combined length by addition. Yet we believe that we can predict. Why?

The success of measurement consists of a correspondence between entities, relations, and operations in the target domain (the domain of logs, chasms, spanning, and log connectors, for example), and entities, relations, and operations in the representation domain (numbers, greater than, addition). That correspondence is the "structure-preserving mapping" that's the heart of RTM. If the structures in the two domains don't correspond in the right way, the answers we get from measurement won't work in the world.

RTM shows that different measurement systems are possible, as long as these correspondences hold up. For example, it would be possible to measure length in such a way that to predict the length of combined logs, one would *multiply* the lengths of the constituent logs, instead of adding them.

How could this work? One needs a special ruler, marked like the one shown in Figure 1.3, at the top. Note the odd feature that the length of no log at all is not 0, but 1. This is required, because combining no log at all with another log does not change its length, and 1 is the measurement that does that in a multiplicative system, while 0 is the measurement that does that in the familiar way of doing measurement using addition. The lower part of the figure shows a worked example: one log measures 4, with the odd ruler; the second measures 8, and (as predicted) the combined log measures 32.

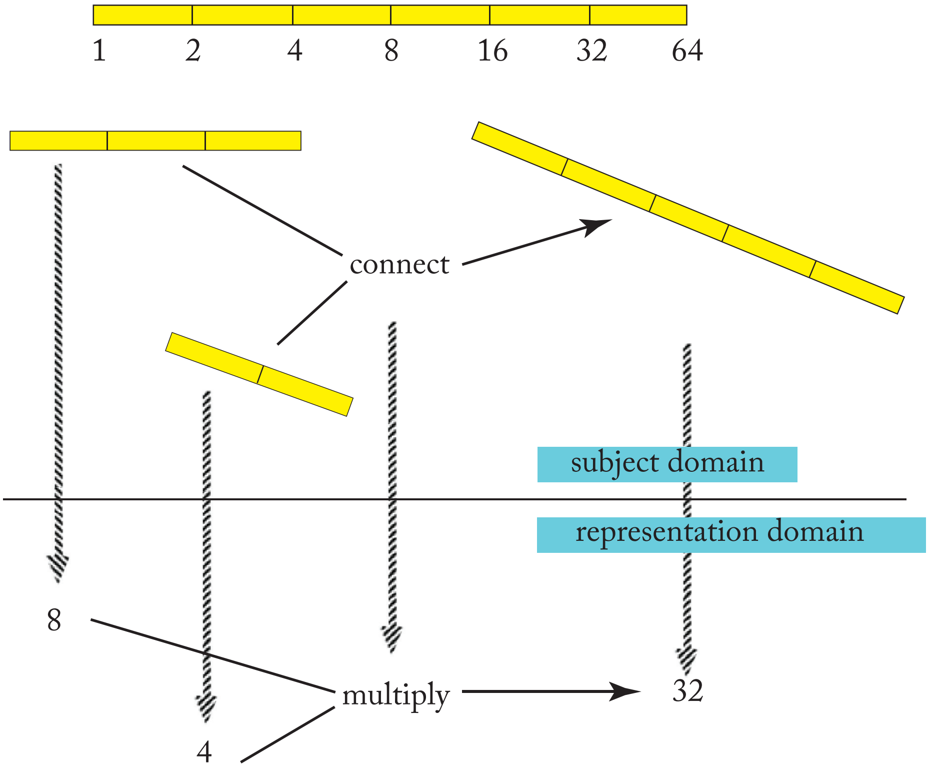

Figure 1.3: Multiplicative measurement of length.

Could one have a system of measurement in which one *subtracts* the lengths of logs to predict their combined lengths? RTM tells us *no*, however strange a ruler we use. The problem is that combining logs is *commutative* where length is concerned: we get the same length if we connect log 1 to log 2, or log 2 to log 1. But the results of subtraction will be different in these cases, so the necessary correspondence does not hold.

RTM deals with numbers and even simpler mathematical systems, like things that can be placed in order, but not used in arithmetic, for example responses chosen from the scale "strongly agree, agree, disagree, strongly disagree." A more general theory of representation needs to relax this restriction, so that we can consider a much wider range of representations. Graphs, sounds, and animations are all possible representations, but aren't simple mathematical objects. We need to allow representation domains that have all these more complex things in them.

Stephen Palmer (1978), in his seminal essay, "Fundamental aspects of cognitive representation", took this step. He analyzed representational systems as containing two worlds, a represented world and a representing world, corresponding to our target and representation domains. And he discussed representing worlds that included such structures as bar charts, or node and link diagrams, and he noted how these systems relate to RTM. Mackinlay and Genesereth (1985) also explored more complex representational systems, followed up by Mackinlay in his dissertation (published as Mackinlay, 1986). While they did not connect their work with RTM, they did develop the principle of corresponding structures, in a different form. They consider *facts* (in a target domain) and statements in a *language* (representation domain). They note that particular structures of facts may not be expressible by statements in a given language, or that expressing some true statements in a language may necessarily also express false ones. These difficulties reflect differences between the structure of the facts and the structure of the language. As in RTM, these structures must match up for the representation scheme to work.

Here's an example of a failure that they use to clarify this point. Suppose one wants to represent relationships among regions in a map, in particular, when one region touches another. One could try to represent each region by a circle, and construct a sentence describing some situation, using the circles, by putting some circles inside others. One could say that region A touches region B, if and only if the circle representing region B is drawn inside the circle for region A, as shown in Figure 1.4.

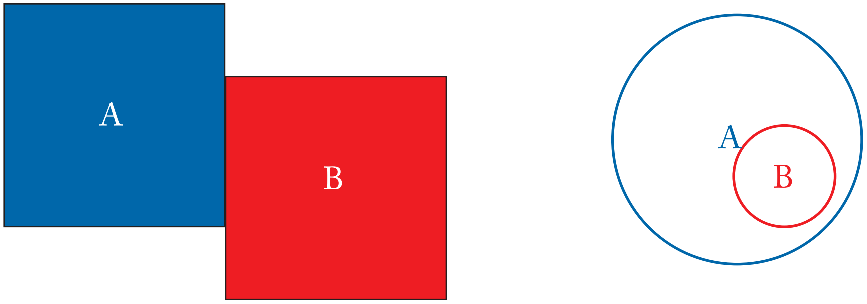

Figure 1.4: Trying to represent touching by nesting.

But this language can't express the facts. In the target domain, if region A touches region B, then region B must touch region A. But in the sentence, only one of these facts can be shown.

In the circle language one has to choose between saying A touches B and B touches A; one can't say both. That's because if A is nested within B, B can't be nested within A, and vice versa. Mathematically, the problem is that touching is a *symmetric* relation A touches B implies B touches A, while nesting is an *antisymmetric* one (A nested in B implies B *not* nested within A).

Box 1.3

One can have fun with Mackinlay and Genesereth's example by changing the use of the circles so as to fix this symmetry problem. Suppose one expresses "A touches B" by requiring *either* that the circle for A is nested in the circle for B, *or* that the circle for B is nested in the circle for A. This revised nesting relation is symmetric, as required.

But Mackinlay and Genesereth point out a further structural problem with the circle language: nesting is *transitive*, while touching is not. That is, if A is nested within B, and B is nested within C, then it follows that A is nested within C. But if region A touches region B, and region B touches region C, it's *not* necessarily the case that region A touches region C.

Does the modified definition of nesting deal with this problem? In part, because the modified relation isn't in general transitive. We could put the circles for A and C next to one another inside the circle for B, as shown in Figure 1.5. Now we have represented the facts that A touches B, and B touches C, but have *not* said that A touches C, because neither is nested within the other.

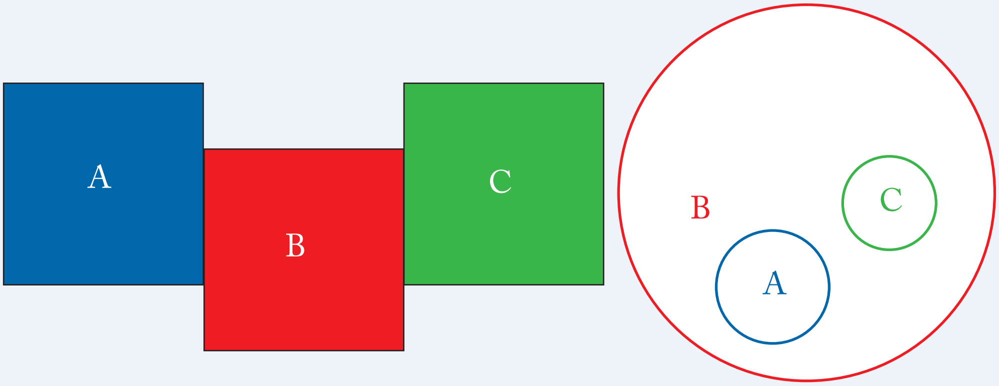

Figure 1.5: Using symmetric nesting to represent touching.

Going further in exploring the modified nesting relation requires that we allow shapes other than circles, so that we can be more flexible in what is nested in what. Doing this allows us to represent the touching in some complex maps, like that of the New England states plus New York, as shown in Figure 1.6.

You can check that all and only the facts about touching are expressed in the diagram using nesting. For example, Maine only touches New Hampshire, because the circle for Maine is only nested within the circle for New Hampshire, and nothing is nested within the circle for Maine.

Does this mean that modified nesting is in fact an adequate representation for touching? No. There are maps it cannot represent. You may want to explore this yourself. Spoiler: the simplest structure that can't be represented with nesting is shown in Figure 1.7.

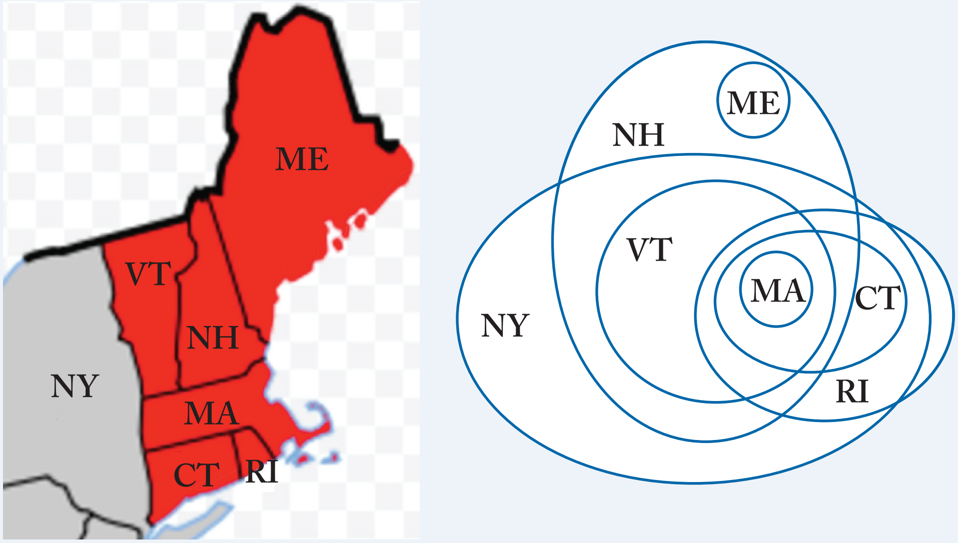

Figure 1.6: Map of New England, and representation of touching using symmetric nesting.

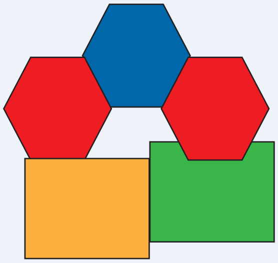

Figure 1.7: A map whose touching cannot be represented with symmetric nesting.

Mackinlay and Genesereth's languages include a wide variety of graphical presentations, more useful than nesting. The point of their analysis isn't to show when representations don't work, but to demonstrate when they do work. They show that a wide range of familiar graphs, like bar graphs and scatter plots, do work to represent a variety of structured data. Such representations can express all the facts of a situation, without implying any false assertions, as required by the principle of structural correspondence.

Mackinlay and Genesereth have taken us far beyond the province of measurement, by considering much more complex representation domains. Mackinlay (1986) makes a further point of fundamental importance: structural correspondence is necessary, but not sufficient, for a representation system to be useful. The added requirement is that it must be possible for someone using a representation to carry out any needed operations, and make any required judgements, easily and accurately.

How might this fail? Mackinlay refers to the work of Cleveland and others (e.g., Cleveland and McGill, 1984) showing that different kinds of graphs place differing demands on the perceptual capabilities of the person viewing them. Figure 1.8 illustrates this point. It shows three bar graphs, drawn differently. All accurately express which of two quantities is larger, because in each case the yellow bar is longer than the red one; bar graphs work by mapping quantities in the target domain to lengths of bars.

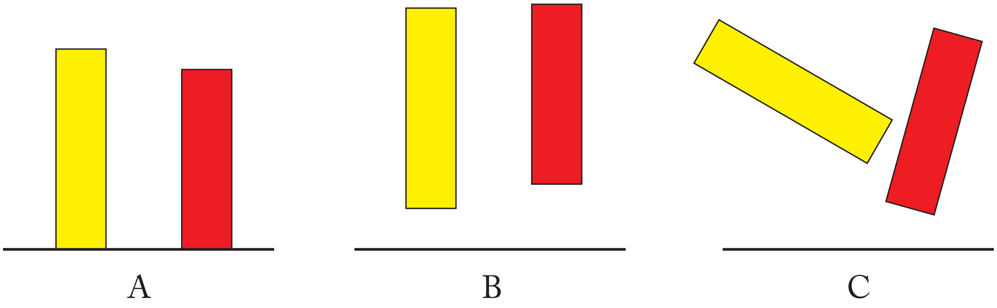

Figure 1.8: Bars that are easily or difficult to compare.

But one can see that these example graphs differ in how easily and accurately the required comparison can be made. When the bases of two bars are aligned, and the bars are parallel, as at A in the figure, the judgment is easy. When the bases are not aligned, and the bars aren't parallel, as at C, the judgement is hard.

Mackinlay calls this further requirement *effectiveness*, adding it to *expressiveness*, the requirement for structural correspondence identified in RTM.

The systems covered by Mackinlay and Genesereth go well beyond the theory of measurement, as we've seen, but there is still more territory to be explored. The languages Mackinlay and Genesereth consider all are static, and all visual. What about animations, and sounds? What does interactivity add? We can frame a still more general theory of representation just by expanding the scope of allowed representational domains to include dynamic structures, interactive systems, and more. This isn't a new theory, but rather the same theory applied to more situations, much as Mackinlay and Genesereth (implicitly) applied RTM to a wider scope, including graphical depictions.

We can further extend the scope of the theory by broadening the idea of the user, in Mackinlay's effectiveness condition. One effect of this is to open the way to consideration of inclusive design, that is, design to support users with a wide range of capabilities. While Mackinlay considers human perceptual capabilities in general, in assessing effectiveness, the logic of his analysis extends equally to the capabilities of particular groups or individuals, which can and do differ. An effective representation must be effective for a particular interpreter, and hence may differ from an effective representation for someone else.

Similarly, without fundamental change to the theory, we can consider other processes than perceptual ones operating on representations. These include cognitive processes: can the user understand the relationships depicted?

It also makes sense to consider computational processes operating on representations, not just operations carried out directly by people. While this needs a more extended treatment than is appropriate here, I've suggested elsewhere (http://comprep.blogspot.com) that computational systems generally provide value because they are representational systems, transforming difficult problems in a very wide range of target domains into tractable problems in computational representational domains. The value of such representations is essentially independent of human perceptual or cognitive capabilities, in general.

I included interactive systems in the list of potential representation domains above. Do these introduce new considerations? To some extent. I'll develop this question when considering our first example, in the next chapter. To foreshadow that discussion, interactive systems include cues for actions in the representation domain, and these cues must be readily interpretable, which does extend the idea of effectiveness somewhat beyond Mackinlay's idea.

1.1 FROM THEORY TO PRACTICE: CREATING A NEW REPRESENTATIONAL SYSTEM, AND USING IT

With this theoretical background in mind, we can outline what is involved in creating a new representational system. We need to have a target domain in mind, since our representational system, to be useful, has to establish a structure preserving mapping between that and some representation domain. But how can we know a target domain, and its structure, without having some kind of representation of it already?

Often, we do use some prior representation in order to know what the structure is. In creating a new representation we are hoping to provide a new representation domain, and a correspondence between that and the target domain, that is easier to work with in some way than what we start with. For example, in representing a collection of numbers as a bar chart, we hope to make it easier to make comparisons. We can do comparisons using the numbers, but using the bars makes it easier. And the numbers tell us enough about the structure of the target domain to allow us to consider alternatives. In working in this way, we are using the layering pattern of representational systems, mentioned earlier, where the representation domain for one system becomes the target domain for another.

Does this mean that there is an infinite regress hiding in the origin of representations? How could there be a *first* representation of something? In some situations the target domain itself can serve as a first representation. In the example of the logs and the chasm, people could accumulate enough experience working directly with logs and combinations of them to discern the structure of the domain. For example, people could see that combining logs is commutative, where what we care about is what gaps the combinations can span.

Another way to get started is by trying out a representation. We may not know up front whether or not a proposed representation actually does capture the structure of the target domain, but experience with it may tell the tale. Some of the examples to come in later chapters have this character. For higher-dimensional structures in space, or musical forms, for example, how well or poorly a proposed representation works may suggest things about the target domain that weren't clear before.

With the target domain in mind, we seek to construct a representation domain, and operations within it, whose structure corresponds to the structure of the target domain, or to what we think we know about the structure. We also have to consider effectiveness: will performing the needed interpretive operations in the representation domain be easy?

Often layering works here, too: if we can map the structure of a target domain onto numbers, we already know lots of ways to represent structures of collections of numbers. And we know that many of these familiar schemes, like bar charts, have reasonable effectiveness.

But, in general, there's no magic here. Indeed, that's why this book is about explorations: there are lots of new domains out there, with new structures, and some may provide useful representations for important target domains.

Once we've created a new representation, how is someone going to learn how to use it? As we've discussed, and as Figure 1.1 shows, representations aren't self-explanatory, considered in isolation.

There's a substantial literature on how people learn to use representations, in the context of math and science learning. Researchers have drawn on a theoretical framework broadly similar to ours, with linked domains. For example, "the most important modeling processes are translations or mappings between contexts" (Lesh, 1981; Goldin and Kaput, 1996) draw specifically on Palmer's two worlds conception (1978).

The picture that emerges is of a *social* process: "the meaning of diagrams, tables, and other displays is established through socially situated efforts to reason with and to interact through artifacts rather than through apprehending self-evident semantic properties of those artifacts (White and Pea, 2011)." That is, people learn to use a new representation by working with it, often along with other people, not from just thinking about the logic of the representation.

The math and science learning literature also includes interesting examples of learners creating new representations. In diSessa et al. (1991), students developed a wide variety of representations of the speed and position of moving objects, using such structures as slanting lines, whose slope represented speed, placed along a horizontal time line. Because these students, sixth graders, did not begin work with a settled understanding of the relationships among time, position, and velocity, their work is an illustration of how trying out representations can be a way to develop one's understanding of the structure of a target domain. Interestingly, working with various representations over a period of days, the students arrived at a graph of position versus time. This arose not as a standard representation provided for them, but as something they evolved from the slanted lines representation, in a social discussion process. Also, interestingly, the students were aware that their representations would not be understood by someone from outside their discussion.

1.2 REPRESENTATIONS AND INCLUSIVE DESIGN

As already discussed, Mackinlay's effectiveness notion can be extended to support people with different capabilities. Whatever operations are required in the representation domain, to accomplish a user's tasks in the target domain, these need to be such that the user can accomplish them easily and accurately. Since people differ in what they can do easily and accurately, it's clear that different representations are required for different people.

Many of the explorations in this book are motivated by this fact. In particular, many of them address the needs of people who can't see well. In many situations the representations in familiar use are visual in character, that is, they require people to perform visual tasks easily and quickly. Plainly such representations aren't useful for people who can't see. The challenge then arises, how can the structure of various target domains, that are usually mapped to visual representation domains, be mapped to representation domains suitable for people who can't see?

Most of the examples in the book use auditory representation domains. There are others, well worth exploring, such as haptic domains that rely on operations of touching or feeling. These aren't explored here, not because they aren't valuable or interesting, but because of a more or less arbitrary choice I've made to focus on representations that can be supported on nearly every machine using a commodity Web browser. There is much more territory than I can explore anyway, so some limitations are unavoidable, and that's one I've chosen.

Of course I can't cover everything within this platform constraint. A very important matter, addressed only very lightly, is that of differences in the cognitive operations required in different representation domains. As already mentioned, in general representations require not only perceptual operations, like comparing the lengths of two bars in a bar graph, but also cognitive operations. I return to this matter briefly in Chapter 2, in considering what has to be understood in using an interactive program.