CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Across the spectrum of human enterprise in government, business, and science, data-intensive systems are changing the scale, scope, and nature of the data to be analyzed. In data-intensive science (Hey et al., 2009), various instruments such as the Australian Square Kilometre Array (SKA) of radio telescopes (www.ska.gov.au), the CERN Hadron particle accelerator (http://public.web.cern.ch/public/en/lhc/Computing-en.html), and the Pan-STARRS array of celestial telescopes (http://pan-starrs.ifa.hawaii.edu/public/design-features/data-handling.html) are complex systems sending petabytes of data each year to a data center. An experiment for drug discovery in the pharmaceutical and biotechnology industries might include high-throughput screening of hundreds of thousands of chemical compounds against a known biological target, or high-content screening of a chemical agent against thousands of molecular cellular components from cancer cells, such as proteins or messenger RNA. Data-intensive science has been called the fourth paradigm that requires a transformed scientific method with better tools for the entire research cycle "from data capture and data curation to data analysis and data visualization." (Hey et al., 2009)

In 2004, the Department of Homeland Security chartered the National Visualization and Analytics Center (NVAC) to direct and coordinate research

Making Sense of Data III: A Practical Guide to Designing Interactive Data Visualizations,
First Edition. Glenn J. Myatt and Wayne P. Johnson.

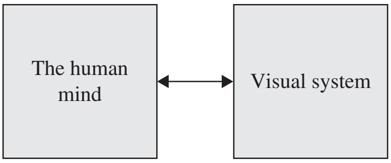


FIGURE 1.1 Intelligence amplified through visual interaction

and development of visual analytics technology and tools. Its major objectives included defining a long-term research and development agenda for *visual analytics* tools to help intelligence analysts combat terrorism by enabling insights from "overwhelming amounts of disparate, conflicting, and dynamic information." (Thomas & Cook, 2005) This has given rise, more broadly, to the emerging field of visual analytics. The field seeks to integrate information visualization with analytical methods to help analysts and researchers reason about complex and dynamic data and situations. But why the emphasis on visualization as a key element in the solution to helping with the problem of data overload?

In 1994, Frederick Brooks in an acceptance lecture given for the ACM Allen Newell Award at SIGGRAPH said:

> "If indeed our objective is to build computer systems that solve very challenging problems, my thesis is that **IA > AI**; that is, that *intelligence amplifying* systems can, at any given level of available systems technology, beat AI [artificial intelligence] systems. . . . Instead of continuing to dream that computers will replace minds, when we decide to harness the powers of the mind in mind-machine systems, we study how to couple the mind and the machine together with broadband channels. . . . I would suggest that getting information from the machine into the head is the central task of computer graphics, which exploits our broadest-band channel."

As shown in Fig. 1.1, to effectively design intelligence amplifying (IA) systems requires an understanding of what goes on in the mind as it interacts with a visual system. Clues about how the mind interprets the digital world come from what is known about how the mind interprets the physical world, a subject that has been studied in vision science.

1.2 VISUAL PERCEPTION

Imagine yourself driving into an unfamiliar large metropolitan city with a friend on a very crowded multilane expressway. Your friend, who knows the city well, is giving you verbal directions. You come to a particularly complicated system of exits, which includes your exit, and your friend says "follow that red sports car moving onto the exit ramp." You check your rearview mirror, look over your shoulder, engage your turn signal, make the appropriate adjustments to speed, and begin to move into the space between the vehicles beside you and onto the exit ramp. Had this scenario taken place, you would have been using visual perception to inform and guide you in finding your way into an unfamiliar city.

The human visual system, which comprises nearly half of the brain, has powerful mechanisms for searching and detecting patterns from any surface that reflects or emits light. In the imagined scenario, the optic flow of information moment by moment from various surfaces—the paint on the road dividing the lanes, the vehicles around you, the traffic signs, the flashing lights of turn signals or brake lights—creates *scenes* taken in and projected onto the retinas at the back of the left and right eyes as upside-down, two-dimensional (2-D) images. The visual system, through various processes executed by billions of highly connected biological computational elements called neurons operating in parallel, extracts information from a succession of these pairs of images in a fraction of a second and constructs a mental representation of the relevant objects to be aware of while navigating toward the exit ramp and their location in the external world.

The perception of a scene from a single moment in time is complex. A 3-D world has been flattened into a pair of 2-D images from both eyes that must be reconciled and integrated with information from past experience, previous scenes, and other sources within the brain to reconstruct the third dimension and generate knowledge relevant to the decisions you are making. There are several theories about how the various perceptual processes work, the representations of their inputs and outputs, and how they are organized. But a generally accepted characterization of visual perception is as stages of information processing that begin with the retinal images of the scene as input and end with some kind of conceptual representation of the objects that are used by thought processes for learning, recall, judgment, planning, and reasoning. This information-theoretic approach divides the general processing that takes place in vision into four stages as shown in Fig. 1.2.

Image-based processing includes extracting from the image simple 2-D features, such as edge and line segments or small repeating patterns, and their properties, such as color, size, orientation, shape, and location.

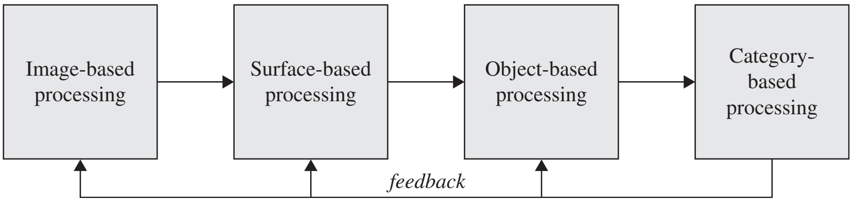


FIGURE 1.2 Information-theoretic view of visual perception

Surface-based processing uses the simple 2-D features and other information to identify the shapes and properties of the surfaces of the objects in the external world we see, and attempts to determine their spatial layout in the external world, including their distance from us. However, many surfaces of objects are hidden because they are behind the surfaces of objects closer to us and cannot be seen.

Object-based processing attempts to combine and group the simpler features and surfaces into the fundamental units of our visual experience: 3-D representations of the objects and their spatial layout in the external world of the scene. The representation of an object is of a geometric shape that includes hidden surfaces, the visible properties of the object that do not require information from experience or general knowledge, and 3-D locations.

Category-based processing identifies these objects as they relate to us by linking them with concepts from things we have seen before or are part of our general understanding of the world, or that are being generated by other systems in the brain such as those processing speech and language. Classification processing uses visible properties of the object against a large number of conceptual patterns stored in our memory to find similar categories of objects. Decision processing selects a category from among the matching categories based either on novelty or uniqueness.

The visual processing just described is a simplification of the process and assumes a static scene, but the world is dynamic. We or the objects in our visual field may be moving. Moment by moment we must act, think, or reflect, and the world around us is full of detail irrelevant to the task at hand. The optical flow, a continuous succession of scenes, is assessed several times a second by small rapid movements of our eyes called *saccades* that sample the images for what is relevant or interesting. In between, our gaze is fixed for only fractions of a second absorbing some of the detail, for there is far too much information for all of it to be processed. The overload is managed by being selective about where to look, what to take in, and what to ignore. Vision is active, not passive. What we perceive is driven not only by the light that enters our eyes but also by how our attention is focused. *Attentional focus* can be elicited automatically by distinct visual properties of objects in the scene such as the color of a surface or the thickness of a line, or by directing it deliberately and consciously. We can intentionally focus on specific objects or areas of the scene relevant to the task *overtly*, through movement of the eyes or head, or *covertly* within a pair of retinal images, through a mental shift of attention.

In the imagined scenario earlier, by uttering the phrase “follow that red sports car,” your friend defined for you a *cognitive task*—move toward an object on the exit ramp—and described the particular object that would become the target of a *visual query* with distinct properties of color and shape to help you perform it. The instruction triggered a series of mostly unconscious events that happened in rapid succession. Based on the goal of looking for red objects along exit ramps and prior knowledge that exit ramps are typically on the outer edge of the highway, the *attentional system* was cued to focus along the outer edge of the expressway. Eye movements, closely linked with attention, scanned the objects being visually interpreted in this region. The early part of the visual processing pathway was tuned to select objects with red color properties. Red objects within the focal area, assuming there were only a few in sight, were identified almost immediately by the visual system and indexed in a visual memory buffer. These were categorized and considered by later-stage cognitive processes, one at a time, until the red sports car was found.

The goal shifted to tracking and following the sports car. Your eyes fixed on the sports car for a moment and extracted information about its relative distance from you by processing visual cues in the images about depth. These cues included occlusion (the vehicles whose shapes obscure other vehicles are in front), relative size (the longer painted stripes of a lane divider are closer than the shorter ones), location on the image (the closer painted stripes are below the farther painted stripes of a lane divider), and stereopsis (differences in location of the same object in the image from the left and right eye that allowed calculation of the object's distance from you). The eyes then began a series of saccades targeting the vehicles in front of and next to you to build up the scene around your path as you maneuvered toward the exit.

Every day as you reach for the handle of a cup, scan the spines of books on the shelves of a library or surf the Web, your eyes and brain are engaged in this kind of interaction and activity to parse and interpret the visual field so that you can make decisions and act. Yet you are mostly unaware of the many complex transformations and computations of incoming patterns of light made by the neural cells and networks of your brain required to produce a visual experience of a spatially stable and constant world filled with continuous movement. Replace the scene of the external world with visual forms that can be displayed on computer screens, and the same neural machinery can be used to perceive an environment of digital representations of data to make different kinds of complex decisions. If the visual forms are carefully designed to take advantage of human visual and cognitive systems, then we will more easily find or structure individual marks such as points, lines, symbols, or shapes in different colors and sizes that have been drawn to support various cognitive tasks.

1.3 VISUALIZATION

The scenario in the previous section used visualization—imagining what was not in sight—to introduce the human visual and cognitive systems. The technical fields of scientific, data, and information visualization and visual analytics use the term *visualization* differently to mean techniques or technologies that can be thought of as visualization tools (Spence, 2001) for making data visible in ways that support analytical reasoning. The essence of this definition includes the person doing the analysis, the user interfaces and graphics that we will call visual forms, and the data. We cannot design effective visualization tools or systems without thinking about the following:

- The analytical tasks, the work environment in which these tasks are done, and the strategies used to perform them
- The content and structure of the visual forms and interaction design of the overall application and systems that will incorporate them
- The data size, structure, and provenance

In the earliest days of computer-supported data visualization, the tasks focused on preparing data graphics for communication. Data graphics were the points, lines, bars, or other shapes and symbols—marks on paper—composed as diagrams, cartographic maps, or networks to display various kinds of quantitative and relational information. The questions included how graphics should be drawn and what should be printed to minimize the loss of information (Bertin, 1983).

With advances in computation and the introduction of computer displays, the focus began to shift to exploratory data analysis and how larger datasets with many variables could be visualized (Hoaglin et al., 2000). Three examples follow. John Tukey and his colleagues introduced PRIM-9 (1974), the first program with interactive graphics for multivariate data that allowed exploration of various projections of data in space up to nine dimensions to find interesting patterns (Card et al., 1999). Parallel coordinates (1990), a visualization tool for multidimensional space, showed how a different coordinate system could allow points in a multidimensional space to be visualized and explored just as 2-D points are in scatterplots using the familiar Cartesian coordinate system. SeeNet (1995), a tool for analyzing large network data consisting of a suite of three graphical displays that included dynamic control over the context and content of what was displayed, was developed to gain insight about the sizes of network flows, link and node capacity and utilization, and how these varied over time (Becker et al., 1999).

The last example in the previous paragraph shows the influence of advances from the human-computer interaction (HCI) community that were made in the 1980s and 1990s in the user interfaces of the three displays of the SeeNet tool and how it changed the way work was done. Instead of using traditional methods of data reduction that aggregated large numbers of links or nodes, averaged many time periods, or used thresholds and exceptions to detect changes, the dynamic controls allowed changes to display parameters that altered the visualizations so all of the data could be viewed in different ways (Becker et al., 1999). The user interface techniques of direct manipulation and interaction, important to the tasks of exploration, had taken priority over the quality of static graphics. For data-intensive analysis, data visualization and user interfaces were converging, and exploration was being done not just by the statistician, but also by domain experts, in this case, engineers in telecommunication responsible for the operations of large networks.

Alongside the new directions in data visualization, the HCI community was taking advantage of advances in computer graphics that included a new understanding of the human-computer interface as an extension of cognition; an expanded definition of data, which included abstract or nonnumeric data; and the emerging World Wide Web. Important new user-interface techniques and visualization tools were introduced that gave rise to the field of information visualization. A sample of these techniques and tools include the following:

- Dynamic queries that could be performed through user-controlled sliders in the user interface instead of through text-based queries, and provided immediate and constant feedback of results through a visual form (Ahlberg & Shneiderman, 1999).
- Techniques to support the need for seeing context and detail together and for seeing different information in overviews than for detailed views.
- A general-purpose framework that used panning and zooming—a form of animation—to see information objects in a 3-D space at different scales (Bederson et al., 1994). Google Maps™ mapping service and Google Earth™ mapping service are examples of this approach.
- Information visualization workspaces that allowed direct manipulation of the content so that the user could focus only on what was relevant, reorganize it into new information, or prepare it for presentation (Roth et al., 1997).

Information visualization had broadened the definition of exploratory data analysis. It now included tasks such as searching, dynamically querying information, grouping and reorganizing data, adjusting levels of detail, discovering relations and patterns, and communication. Interactive visual forms could operate with numeric or abstract data, allowing for the results of statistical calculations or data-mining computations to be linked to the information objects from databases or data tables or to other complex objects such as chemical structures from which they were derived.

The past couple of decades have also seen the emergence of data-intensive science. Projects in the physical and life sciences generate large amounts of data that originate from a variety of sources and flow into data centers from complex collections of sensors, robotics, or simulations from supercomputers or grid computing. Data capture, curation, and analysis are done by teams of individuals who are often geographically dispersed and have expertise in IT, informatics, computational analysis, and a scientific discipline. This has given rise to high-throughput data analysis and exploratory tools.

The data comes in all scales. It might be a large dataset with values from a single experiment or a family of datasets from related experiments. For example, the National Institutes of Health (NIH) has carefully defined sets of rules and procedures—protocols—for conducting experiments that allow chemical compounds screened against a set of cancer cell lines to be compared with the values of data from the screens of other cellular parts—for example, genes, messenger RNA, or micro RNA—across the same set of cancer cell lines using microarray technology. The ability to integrate data across experiments provides insight into various mechanisms involved in cancer.

The data to be analyzed can come from one of several stages of processing. For example, in a microarray biology experiment, the amount of each gene expressed in a cell can be measured by the intensity of light at a point on a microarray where the mixture containing the gene has been spotted. From the microarray, a machine produces a digitized graphic image. Image analysis converts the digital image to a matrix of numbers. The image analyst might explore the analog signals from the laser scanner, the statistician might explore the matrix of numbers, and the biologist might explore the genes of a particular group discovered by clustering or the factors of a principal component analysis generated by the statistician.

A broad collection of computational tools and algorithms are available to support high-throughput analysis tasks, which include many of the same tasks described for data and information visualization. These tools and algorithms are drawn from classical statistics, machine learning, and artificial intelligence (AI).

The data is distributed across a network and stored in a variety of formats under control of different data-management systems. Some of it may contain data called *metadata*, that describes the raw data. For example microarray experiments are recommended to contain the minimum information about a microarray experiment (MIAME) needed to interpret and reproduce the experiment. The information includes the raw data, the normalized data, the experimental factors and design, details about each item in the microarray, and the laboratory and data processing protocols (FGED, 2011). Other associated data will also need to be linked. For example, the IDs of genes that reside in the headings of a microarray matrix of numeric expression values can be used to retrieve information about their function or their sequences.

Once again, the definition of exploratory data analysis has broadened. The data to be analyzed is no longer only within a single file or data table. Even if the primary focus is on numeric data, the objects or observations from which the numbers were derived—abstract data—and information about the provenance of the data or details about the experiment are important aspects of exploration that are integrated into the user interface with the visual forms. The data may require stages of processing where the output of any stage may be of interest. It may require subsets to be retrieved through queries to remote servers. The computational tools may run on the same machine as the visual tool, as a service accessible via the Internet, or as a large-scale computation distributed over many computers. Teams of experts with different skills—for example, a molecular biologist, a bioinformatics specialist, and a computational biologist—may be required to analyze the data.

The fields using computer graphics for visual forms—data graphics, information visualization and user interfaces—have rich traditions of design. Interaction techniques are increasingly being added to data graphics, new visualizations are being invented to handle the much larger scale of data being generated by data-intensive methods, and a greater variety of information is being linked into exploration. As this continues, the boundaries between the three kinds of visual forms within a visualization application will eventually disappear if it is designed without seams. The visualization systems for exploration in data-intensive environments will benefit from the considerable body of research and practical knowledge on design that comes from these traditions, and we will draw on each of them in this book.

1.4 DESIGNING FOR HIGH-THROUGHPUT DATA EXPLORATION

Whereas the concern of the analyst is exploring the data, the concern of the designer is to create a visualization tool through which the analyst "sees" the data but not the tool. For the work the tool is designed to support, interaction with the tool should be engaging and not interrupt the flow of thought. This is the ideal for usability. The fields of human-computer interaction (HCI) and interaction design (ID) have spent decades studying how to achieve this in interactive computing systems and various products, technologies, systems, and services. For aspects of the visual forms that are static, rules and principles that have evolved from centuries of print tradition can also inform design.

1.4.1 The IA (Intelligence Amplified) System

A conceptualization of the IA system for high-throughput data exploration, which is the focus of our design, is shown in Fig. 1.3.

The analyst, referred to in design as the *user*, is the *primary* focus. For those coming from engineering or computer science without a design background, it is important to recognize the shift in perspective from technology to user. To a designer, technology is a design material—something usually selected later in the design process to solve the needs of the user after they are understood. The visual and cognitive systems of the brain have already been introduced. They are the critical part of the overall IA system and these systems interpret and act on the visual information that is a source for analytical thinking. Viewed as information-processing systems, the visual and cognitive systems that comprise the mind have constraints that must be considered in design.

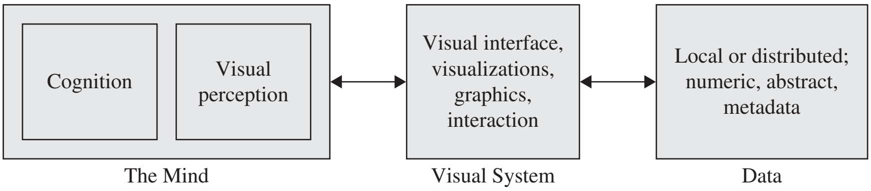

Cognition
Visual perception
Visual interface, visualizations, graphics, interaction
Local or distributed; numeric, abstract, metadata
The Mind
Visual System
Data

FIGURE 1.3 The IA (intelligence amplified) system

The *visualization system* is a computational system capable of interacting with the user through high-resolution displays and interaction devices, and of accessing data from one or more sources on the same machine or on the Internet. It might be implemented as a Web browser running on a PC or a handheld tablet, the client application running on the laptop of a distributed system, or a set of components that controls user interaction with a wall-mounted display.

The visual representation displayed by the visualization system is generically called *visual forms* (Card et al., 1999, p.17). Visual forms are compositions of user interface *components* and *controls*, *graphics* of numeric data, or *visualizations* of abstract data. All interact with the user using the same display and interaction devices and are drawn in a visual language of graphical elements—points, lines, planes, volumes, and so on—with visual properties such as position, color, shape, size, orientation, and texture. The graphical elements are designed to relate to one another in a way that communicates information. For example, a visual form might be a scatterplot with sliders to set the limits on the x-axis and y-axis and filter the points displayed in the plotting region, or an interactive visualization showing the groups of a clustering algorithm, each of which can be selected to retrieve more information about the elements in the group.

The data being analyzed and explored by the users of the visualization system can only be accessed through the visual forms. How and where the visual form is stored is irrelevant to the user but of great importance to the designer. Access to all or part of the visual form comes at a cost in time: data stored locally on the same machine that is generating the display is generally accessed much faster than data stored at a remote site. Certain visual forms may require significant computational time to generate. The response time for user actions affects the interaction and must be taken into account during design.

1.4.2 Design

Every product, physical or not, has an interface. The *interface* is the point of interaction. Teapots have handles which are part of the physical form of the container. Windows have latches to help you lock them. Figure 1.4 shows the control panel of a dishwasher with buttons to let you choose how to wash and dry dishes. The panel has been designed to access the hidden functions programmed into the electronics of the machine. By looking at and

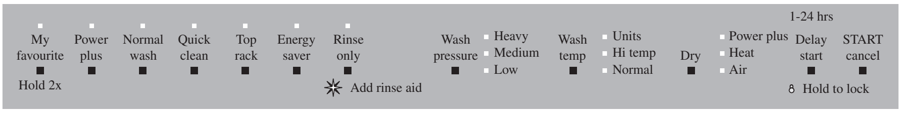


FIGURE 1.4 Dishwasher control panel

experimenting with the interface, we begin to develop a *mental model* of what functions the dishwasher provides, how it works, and how to interact with it.

Because of the layout of the visual elements, we unconsciously perceive a hierarchical organization: starting from the left there is a cluster of seven groups, followed by a cluster of three groups, and then two groups. Every group has text with a black square below; some groups have one white circle, some have three white circles with text to the right, and some have no white circles. What is the meaning of the various combinations of text, black squares, and white circles?

From prior experience with these kinds of panels, the groups leave the impression of buttons. The text of some buttons use terms familiar to anyone who works in the kitchen such as “wash,” “clean,” “wash temperature,” “dry,” or “heat.” Every group has text and a black square; these must be buttons. Because the text in the cluster of seven buttons seems to relate to how dishes get washed—“normal wash” or “quick clean,” these must be buttons that represent a way to set the method for cleaning dishes. The organization in the cluster of three groups is different. If the text and black square represent the function, then maybe the text and white circles are ways to vary that function by degree. The wash temperature is “sanitize,” “hi temp,” or “normal.” Sanitize is not normally associated with temperature, but because the white circle with the “sanitize” text is above “hi temp,” and the white circle with “normal” text is below, “sanitize” must mean very high temperature. Seven buttons relate to a way of cleaning dishes and three to setting different degrees of water temperature, pressure, and drying temperature, but how do we make it work? How are the selections for water temperature and pressure, and drying temperature made? Can “energy saver” and “top rack” be selected simultaneously? Does “energy saver” have any effect on “rinse only”? Which combinations make a difference?

Even in this simple interface, you can see the design questions about how the visual form of the interface relates to function, how the interface reveals and constrains the actions that can be performed, and what the sequence and flow of the interaction should be to support the tasks. No matter how complicated interactive visual forms become, they play the same role for the analyst in their domain as the dishwasher panel does for the cook in the kitchen. The visual forms that the visualization system presents is the only way to see and initiate actions that transform the data, but, as we have seen in the previous example, the design of these forms is critical to how quickly and easily they can be interpreted and understood. So how do we begin to design the interface and interaction with the visualization system?

Interaction design is a highly dynamic process that consists of four basic steps: identify the user’s needs, develop alternative designs, prototype, and evaluate. Although it is presented as a sequence of steps, what is learned in each step provides feedback that may change the result of another step. For example, during prototyping, insight may be gained when converting a low-fidelity prototype—for example, one implemented on paper or a PDF document—to a high-fidelity working prototype that reveals software and hardware bottlenecks requiring changes to the requirements. (Note: Because we are focusing only on the design of the system as it pertains to the visualization of data, we will ignore those aspects of design that would be addressed if a commercial product were being created. These include the context in which it will be used and other aspects of design that include standards, integration with other systems, and portability.)

1. Identify the user's needs, and establish initial requirements. Getting a feel for the work and how the system may be used requires a thorough understanding of the following:

- Who will use it
- The driving problem behind the analysis
- The analysis tasks, and the concepts and actions needed to perform them
- The source, content, and processing of the data
- The usability goals as they relate to user performance

For many data-intensive projects, there may be several people who directly use the system: an informaticist who manages the systematic collection and organization of data so that it can be retrieved or searched, a computational specialist who is trained in statistical and data-mining methods, and a domain expert with knowledge about the relevant subject area such as biology, marketing, telecommunications, or finance. If, for example, a system is being designed to explore the integrated data of biological targets or chemical compounds generated by high-throughput screening or microarray technology in order to study cellular mechanisms of cancer, the users might be molecular biologists, medicinal chemists, and cheminformatics or bioinformatics experts and the domains would be molecular biology, medicinal chemistry, computational biology, bioinformatics or cheminformatics, and statistics.

Becoming familiar with the domain implies becoming immersed in the field. It may include reading textbooks, technical material, or important papers that have been published; interviewing and talking with experts in the field; or attending a conference or workshop. The ideas for design come from a depth of understanding. An understanding of the field provides a foundation for seeing through the users' descriptions about the steps they take in performing their tasks or the feedback they provide during evaluation to intent: what the user is really trying to accomplish. With many experts, the intent may be buried from years of practice so that it is no longer noticed and cannot be articulated. Knowing the intent or goal that the user is trying to accomplish allows alternative designs to be considered that may do things in a different but more efficient way.

Usability, a term used by designers, is a way to think about the usefulness of the analysis system with which the user will interact. Most usability goals include practical measures for assessing them. The goals include the following (Shneiderman & Plaisant, 2010):

- *Time to learn*. How long does it take to learn to do the tasks? The length of time it takes to learn is more of a problem for systems that are used infrequently than it is for a system that will be used daily.
- *Performance*. How long does it take to do each task once the user is proficient?
- *Accuracy*. How many and what kinds of errors are made? Errors refer to any action that does not accomplish what the user intended.
- *Memorability*. How easy is it to remember how to use the system the next time the user interacts with it, for instance, after a day or a week? A complex system that is used infrequently can be as frustrating as a simple system that is used frequently but lacks keystroke shortcuts or functions that allow expert users to adjust parameters. There are trade-offs to consider.
- *Satisfaction*. Did users respond positively or negatively to specific visual forms or areas of the user interface? How did they experience the system as a whole? Qualitative assessments not only provide insight into specific aspects of the design, but psychology research shows that emotions affect how the mind solves problems. Positive experiences open our minds and make us more creative and better able to learn. Negative experiences narrow our thoughts to the problem at hand (Norman, 2004).

The outcome of studying users and their work leads to preliminary requirements that are stable enough to begin design and a set of benchmark analysis tasks that can be used for prototype evaluations.

2. Develop alternative designs. For many software developers or programmers, the first step in design is to begin by sketching ideas about what the interface should look like. However, designers begin by designing a conceptual model focused on the tasks and goals of the user that will, in turn, guide the physical design that includes the details of the interface and interaction.

A *conceptual model* is a simplified, high-level model of the system that designers want to communicate to the users to make it easier for them to understand and use. The interface will be designed to convey and support the conceptual model. Preferably the conceptual model is drawn from the task analysis because the task descriptions contain important words that identify the concepts that users are most familiar with; but the conceptual model could also be based on metaphors or analogies. A widespread example is the office metaphor first introduced in the Xerox Star and now widely used in the Macintosh and Windows PCs. This conceptual model includes a desktop, folders, files, pictures, notes, mail, and other objects. Folders or files can be created or deleted. Files are stored in folders. A file can be edited and its content copied onto a clipboard and pasted into another file. The familiarity of the user with physical objects from the office (desktop, file cabinets, folders, files) allows the actions (move, cut, paste, select, edit) associated with a visible representation of the object on the display to make sense.

The most important component of the conceptual model is the object-action model, which identifies each conceptual object that will be visible in the interface, and each object's actions, attributes, and relationships to the other objects.

Two other areas must be considered in the design that may affect the conceptual model: the style of interaction and the type of interface. These are outside the scope of this book, but we introduce them briefly here and refer you to the end of the chapter for sources of further reading about these areas.

An interaction style characterizes how the user interacts with the system and can be categorized as follows (Sharp et al., 2007):

- *Command*. The user directs the system by entering text in a command language, speaking, selecting menu items from a menu tree, and so on.
- *Conversational*. The system has an ongoing dialog with the user through speech or text.
- *Direct manipulation*. The user initiates actions by selecting an object that is displayed on the screen and choosing an associated action; this style is also known as point-and-click.
- *Exploration*. The users explore a virtual or physical space, for example, virtual reality or smart rooms.

We will assume direct manipulation for the visualization system because it is most widely in use. The Protovis visualization toolkit we will introduce in Chapter 5 for creating visualizations in a Web browser supports direct manipulation.

Many new interface types have been invented recently and are becoming available as design material (Sharp et al., 2007). However, these are not widely in use. We will assume that visualization systems have a presentation layer or client application that runs on a desktop PC or laptop with a high-resolution display, a keyboard for entering text, and a mouse or touchpad as a pointing device.

3. **Prototype**. Prototypes are not a replacement for analysis and design. They are visualizations of some or all of the requirements, benchmark tasks, and conceptual model—a way to see what has largely been verbal descriptions of the tasks to be performed and the steps to perform them that were developed in the previous stages of design. For individuals participating in design, they are helpful for thinking through the details of interface and interaction, for quickly generating designs that can be assessed and incorporated or discarded, and for deciding which alternatives are best to advance. The type of prototype and its role changes as the design process progresses. Prototyping is highly iterative.

At the beginning of physical design, *low-fidelity prototypes* are used to explore designs. Low-fidelity prototypes include sketches of single screens with the user interface components and controls along with visualization or data graphics, paper prototypes, storyboards of task sequences, or high-quality digital sketches exported from illustration drawing or slide presentation software applications. Using combinations of these approaches is also effective. Whatever method is used, it should support the ability for users to physically interact with the prototype to execute limited tasks selected from those developed in task analysis.

As the physical design progresses, *high-fidelity prototypes* are used to explore issues related to critical areas or to develop a working prototype with interaction sequences and flows. The working prototype can evolve into a proof-of-concept system that can be demonstrated outside the design team. High-fidelity prototypes take longer to build, so they should be used for areas of the interface where the design is considered stable and changes are infrequent.

4. Evaluate. The goals for evaluation for design differ from the goals of evaluation that are intended to assess usability. Evaluation for design is intended to provoke discussion about better ways to the structure the system, or to uncover tasks that aren't necessary or may be more or less important than initially thought. The evaluation stage is iterative and must be a fundamental part of the design process from the beginning.

Usability testing, on the other hand, is done in the late stages of development. It measures the users' performance on a set of predefined tasks. It is intended to uncover small problems or areas that are found to be difficult to understand. The changes made are to polish the interface and improve the interaction to refine the product. If the prototyping has been done well, there should be no major surprises.

1.4.3 Data

Data is captured or collected from sensors and sources that range from arrays of charge-coupled devices (CCDs) in telescopic cameras to electronic cash registers in retail stores. The data that is collected—*raw data*—is stored in many ways: into files in proprietary or standard formats, with or without metadata; and into all sorts of databases with various schemas. Before trying to use the raw data for visualization, it is helpful to transform it into a standard *data table* as shown in Fig. 1.5. The data-processing pipelines that convert raw data to data tables depend on the kind of data collected. For example, preparation for the mining and analysis of numeric data often includes cleaning, transformations, reduction, and segmentation, which results in tabular data (Myatt & Johnson, 2009).

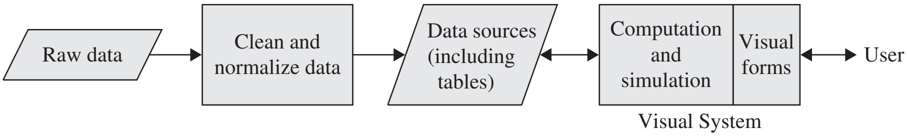


FIGURE 1.5 Data-processing pipeline

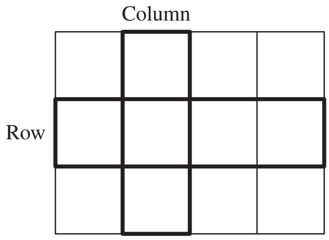


FIGURE 1.6 A simple table and its nomenclature

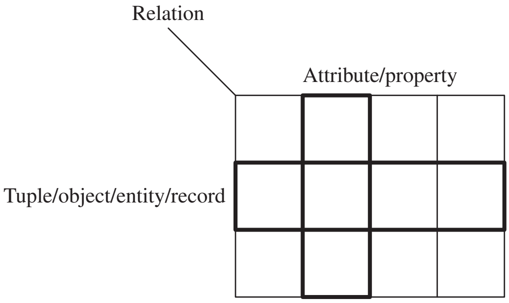


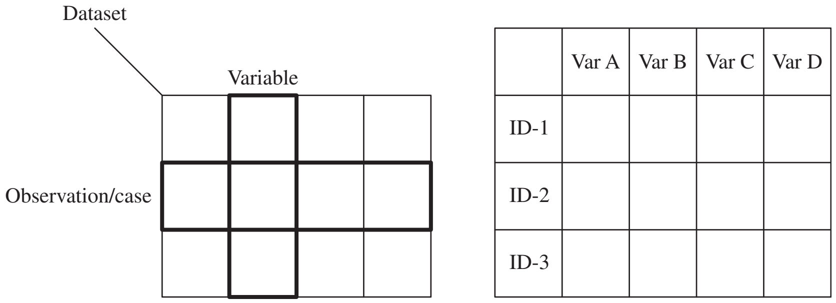
FIGURE 1.7 A relational table and its informal nomenclature

Different methodologies and techniques for data preparation are used in various scientific disciplines, computer science, and statistics (Myatt & Johnson, 2009). When referring to tables, different terms are used for similar concepts or the same term for different concepts. Spreadsheet, row, column, entity, object, relation, table, tuple, record, attribute, property, dataset, case, observation, variable, matrix, and metadata are terms used to describe different perspectives on what is commonly understood as a table of rows and columns as shown in Fig. 1.6. As a guide through the literature and to clarify our use of the term *data table*, we provide a summary of the different ways terms are used and a definition.

In relational databases, data is logically stored in tables of rows and columns. The terms *relation*, *tuple*, and *attribute* are from the mathematical theory of relations that underlie the relational model developed by E. F. Codd. A tuple represented a thing or object in the world and its associated attributes. For example, a machine part in an inventory system was a tuple, and its serial number, name, size, and quantity were attributes. A relation was a set with special properties: all tuples in the set were of the same type, they were unordered, and there could be no duplicates (Codd, 1990). The relational model is at the heart of relational database management systems. Figure 1.7 shows the mathematical terms and others that are commonly used today.

In statistics, *dataset* refers to a collection of information often in a tabular format. For example, the dataset might be a collection of data on patients or cars. The patients or cars are objects. In the dataset, each item in the collection is an observation, and there may be many *observations* for a particular object. The observations may be described in various ways. For example, a car has a vehicle identification number, a manufacturer's name, and a weight. Each of these features describing the car is a statistical *variable*. The variables play a role similar to the one attributes play in database relations. Smaller datasets are often stored in files. In these files, the first row is often made up of the names of column headings, and the first cell of each row is an ID of the object whose values are represented by the remaining cells in the row. The terminology used in statistics is shown in Fig. 1.8.

In visualization, the dataset of statistics is called a data table, but the variables are rows, and the cases are columns as shown in Fig.1.9 (Card et al., 1999). Jacques Bertin, whose work we will study in Chapter 3, uses this orientation of the data table, but he refers to variables as *characteristics* and cases as *objects*. The Protovis toolkit that we will explore in Chapter 5 also uses this orientation of the table and expects input to the visualization pipeline as a "variables by cases" array.



	Var A	Var B	Var C	Var D
ID-1				
ID-2				
ID-3				

FIGURE 1.8 A dataset and its nomenclature; a dataset with row and column headings

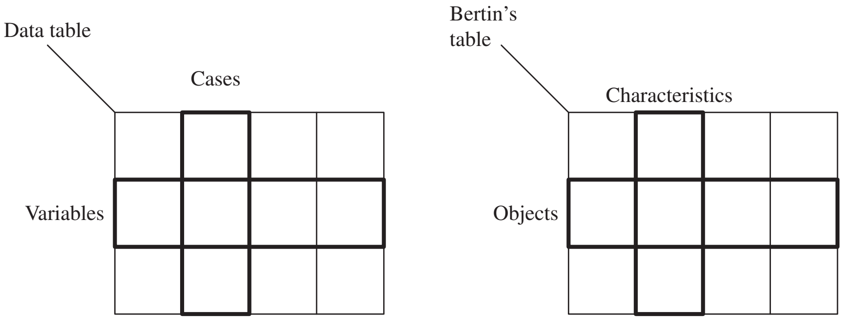


FIGURE 1.9 A data table and Bertin's table with terminology

The definition of data table used in this book adheres to the one commonly used in statistics and data mining: the observations are rows, and the variables are data columns. If datasets include row and column headings, they are treated as metadata. If the first column contains the IDs of the observations and these IDs are not considered values of the dataset the ID values are treated as metadata.

1.5 SUMMARY

Data-intensive systems are changing the scale, scope, and nature of the data to be analyzed. The human visual system, which comprises nearly half of the brain, has powerful mechanisms for searching and detecting patterns. This capability has given rise to the field of visual analytics, which seeks to exploit these capabilities.

To effectively design IA (intelligence amplified) systems requires an understanding of what goes on in the mind as it interacts with a visual system. Information-based theories divide visual perception into four stages that transform the image in the retina of the eye to concepts that we understand. In the image-based stage, simple features are extracted. In the surface-based stage, the simple features and other information are used to find the surfaces of the objects in the external world. In the object-based stage, the simpler features and surfaces are combined and grouped into 3-D representations of objects and their spatial layout. In the category-based stage, the objects are identified, classified, and linked to concepts we have seen before or are part of our general understanding of the world.

The visual exploration of data is an interaction between user and system. We cannot design effective visualization tools or systems without understanding the work, the work environment, the form and content of the visual representations, and the data. Designing visual interactions is a highly iterative process that consists of identifying the user's needs, developing alternative designs, prototyping, and evaluation. This results in visual systems that, even if complex, are understandable and allow the tasks they were designed to support to be efficiently performed.

1.6 FURTHER READING

There are several references that go more deeply into the topics introduced in this chapter.

- The Foreword by Gordon Bell and the overview of eScience by Jim Gray in *The Fourth Paradigm: Data-Intensive Scientific Discovery* provide further background on data-intensive science and the challenges involved in capturing, curating, and analyzing data (Hey et al., 2009).

- The idea of AI was presented in a lecture by Frederick Brooks given in acceptance of the ACM Allen Newell Award at SIGGRAPH in 1994 (Brooks, 1996). The lecture emphasized the role of computer scientists as makers of tools and the importance of using an interdisciplinary collaboration on a real and complex problem as a driving problem for research. It also provides insights into issues that arise when doing user-centered design.
- The first chapter in *Readings in Information Visualization* (Card et al., 1999) provides a detailed overview of information visualization. This is essential reading.
- Chapters 1, 3, and 5 of *Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining* (Myatt, 2006) provide an overview of data preparation and the use of statistics in exploratory data analysis and data-mining applications.
- The first chapter in *Vision Science: Photons to Phenomology* (Palmer, 1999) gives an introduction to vision science that includes a description of the problems in scene recognition, the human visual system, and visual perception.
- Two chapters of *Illuminating the Path: the Research and Development Agenda* for *Visual Analytics* provide helpful background. In the context of providing support for analyzing security threats to a nation, Chapter 2 discusses the science of analytical reasoning, and Chapter 3 discusses various visual representations and interaction technologies.