# Preface

Geographic information systems (GISs) are computer-based information systems that are able to capture, model, store, retrieve, share, manipulate, analyze, and present geographically referenced data. This book is about the technology, theories, models, and representations that surround geographic information and GISs. This study (itself often referred to as GIS or *geographic information science*) has emerged in the last two decades as an exciting multi-disciplinary endeavor, spanning such areas as geography, cartography, remote sensing, image processing, environmental sciences, and computing science. The treatment in this text is unashamedly biased toward the computational aspects of GIS. Within computing science, GIS is a special interest of fields such as databases, graphics, systems engineering, and computational geometry, being not only a challenging application area, but also providing foundational questions for these disciplines.

The underlying question facing this multidisciplinary topic is "What is special about spatial information?" In this book, we attempt to provide answers at several different levels: the conceptual and formal models needed to understand spatial information; the representations and data structures needed to support adequate performance in GISs; the special-purpose interfaces and architectures required to interact with and share spatial information; and the importance of uncertainty and time in spatial information.

The task of computing practitioners in the field of GIS is to provide the application experts, whether geographers, planners, utility engineers, or environmental scientists, with a set of tools, based around digital computer technology, that will aid them in solving problems in their domains. These tools will include modeling constructs, data structures that will allow efficient storage and retrieval of data, and generic interfaces that may be customized for particular application domains.

The book inevitably reflects the interests and biases of its authors, in particular emphasizing spatial information modeling and representation, as well as developing some of the more formal themes useful in understanding GIS. We have tried to avoid detailed discussion of particular currently fashionable systems, and concentrate instead upon the foundations and general principles of the subject area. We have also tried to give an overview of the field from the perspective of computing science.

Not every topic can be covered and we have deliberately neglected two areas, leaving these to people expert in those domains. The first is the historical background. The development of GIS has an interesting history, stretching back to the 1950s. Readers who wish to pursue this topic will find an excellent introduction in Coppock and Rhind (1991), and more in-depth perspectives from many of the pioneers of GIS in Foresman (1998). The other area that is given scant treatment is spatial analysis, which requires specialized statistical techniques and is judged to be specifically the province of the domain experts. Introductions to spatial analysis include Unwin (1981), Fotheringham et al. (2002), and O'Sullivan and Unwin (2002). The bibliographic notes in Chapter 1 provide further references to texts on specific aspects of spatial analysis.

## WHO SHOULD READ THIS BOOK

This book is intended for readers from any background who wish to learn something about the issues that GIS engenders for computing technology. The reader does not have to be a specialist computing scientist: the text develops the necessary background in specialist areas, such as databases, as it progresses. However, some knowledge of the basic components and functionality of a digital computer is essential for understanding the importance of certain key issues in GIS. Where some aspect of general computing bears a direct relevance to our development, the background is given in the text. This book can be used as a teaching text, taking readers through the main concepts by means of definitions, explications, and examples. However, the more advanced researcher is not neglected, and the book includes an extensive bibliography that readers can use to follow up particular topics.

## CHANGES TO THE SECOND EDITION

The second edition of this book was written with the aim of making the book more accessible to a wider audience, at the same time as retaining the core of tried and tested material. Chapters 1–6 have been extensively revised, updated, and reformatted from the first edition, although in a fast moving high-technology area like GIS it was encouraging to find that these fundamental aspects of GIS have remained largely unchanged. Chapters 7–10 present almost entirely new material, covering GIS architectures, GIS interfaces, uncertainty in geospatial information, and spatiotemporal information systems. The bibliography, index, and all the diagrams have also been completely revised.

In addition to the changes in content, we have tried to produce a more attractive and readable format for the book. The following section contains more details on the formatting conventions used in this book and on the structure of the book. The spelling, grammar, and usage in second edition has also changed, from British to American English. We hope that this change will further improve the accessibility of this book to an international audience.

## FORMATTING USED IN THIS BOOK

Several formatting conventions, new to the second edition, have been used in this book. Material that is relevant to the main themes in the text, but not essential to the reader, is included in gray inset boxes at the top of a page. Typically insets contain more challenging material, and provide some background to each topic, as well as references and links, which readers may wish to follow-up. A list of insets can be found on page xi. Every chapter begins with a brief summary, outlining the major ideas in that chapter and highlighting some important terms introduced in the chapter. At its close every chapter ends with itemized bibliographic notes, providing some key references that readers can follow up. The section numbers alongside the bibliographic notes refer to the relevant sections in the main text.

Throughout this book, we have used margin text to allow rapid reference to important terms. When an important term is first defined or introduced, that term will appear in the margin. A corresponding entry can be found in the index, with the page reference in bold typeface. This enables the reader to use the index rather like an extensive glossary of terms used in this book. Each index term has at most one bold typeface page reference, and a term can be rapidly located within a page by finding the corresponding margin entry. In addition to normal- and bold-typeface index entries, those index entries that appear in italics refer to terms that appear within a gray inset box.

## STRUCTURE OF THIS BOOK

Figure 0.1 indicates the overall structure of interdependencies between chapters. Readers may find it helpful to refer to Figure 0.1 to tailor their use of this book to their own particular interests.

*Chapter 1*: Motivation and introduction to GIS; preparatory material on general computing.

*Chapters 2–3*: Background material on general databases and formalisms for spatial concepts

*Chapters 4–6*: Exposition of the core material, forming a progression from high-level conceptual models, through representations and algorithms, to indexes and access methods that allow acceptable performance.

*Chapters 7–8*: Discussion of the types of system architectures and user interfaces needed for GIS.

*Chapter 9*: Introduction to spatial reasoning theory and techniques, with particular focus on reasoning under uncertainty.

*Chapter 10*: Introduction to temporal and spatiotemporal information systems.

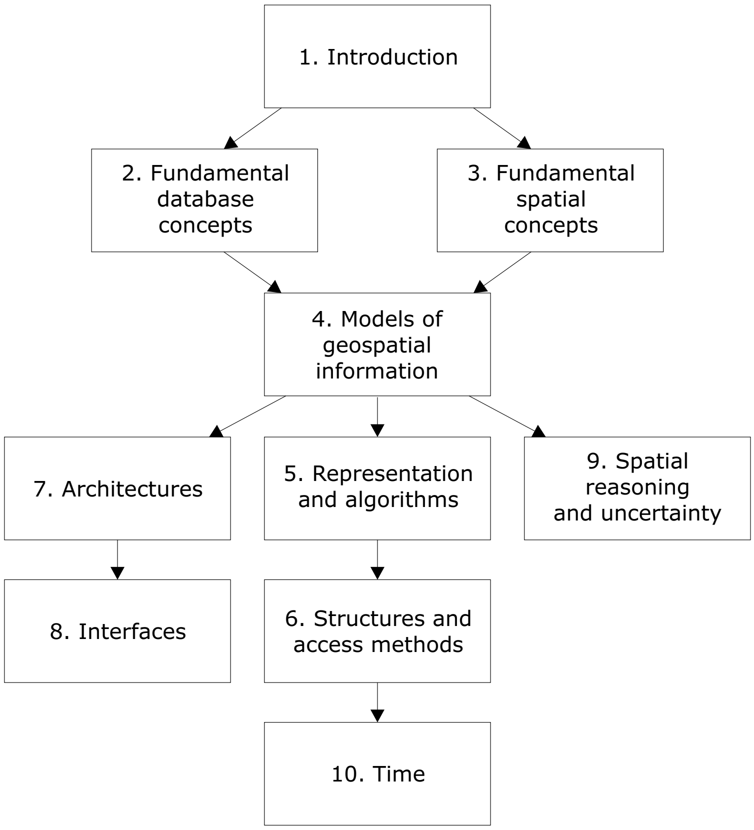


**Figure 0.1:** Relationships between chapters

## ONLINE RESOURCES

The website that accompanies this book can be found at:

http://worboys.duckham.org

The resources at this site are constantly under development, but include resources such as sample exercises, lecture slides and notes, open-source computer code, sample material, useful links, errata, and contact information. We, the authors, welcome suggestions from readers as to resources that we should include on the website, or indeed any feedback or comments on the book itself. We can be contacted on email at gisacp@worboys.duckham.org; other up-to-date contact information can be found on the website.

# Contents

# 1 Introduction

**Summary**

*A* ***geographic information system*** *(GIS) is a special type of computer-based* ***information system*** *tailored to store, process, and manipulate* ***geospatial*** *data. This chapter sets the scene, describing what a GIS is and giving examples of what it can do. At the heart of any GIS is the* ***database****, which organizes data in a form that is easy to store and retrieve. The chapter also describes some of the* ***hardware*** *technology surrounding GIS, including computer processors, storage devices, user input/output devices, and computer networks.*

"What makes GIS special?" Most people who work with geographic information systems have asked themselves this question at one time or another. This chapter starts to answer the question by describing the field of GIS against the general background of computing and identifying what distinguishes geographic information systems from other information systems. First, we define the terms "information system" and "GIS," and outline the main components of a GIS (section 1.1). Then, in section 1.2, we look at some example applications that illustrate what a GIS can do, and provide a motivation for studying this topic. The most important technology underlying GIS is the database system. Databases are introduced in section 1.3 along with an outline of the key features that are distinctive in geospatial databases. The chapter concludes with an overview of the basic computing hardware common to all GISs (section 1.4).

## 1.1 WHAT IS A GIS?

A good starting point for defining a geographic information system, is to look at a general definition of an *information system*. An information system is an association of people, machines, data, and procedures working together to collect, manage, and distribute information of importance

information system

to individuals or organizations. The term "organization" is meant here in a wide sense that includes corporations and governments as well as more diffuse groupings, such as global colleges of scientists with common interests or a collection of people looking at the environmental impact of a proposed new rail-link. The World Wide Web (WWW) is an example of an information system. The WWW comprises data (web pages) and machines (web servers and web browsers), but also the many people across the world who use the WWW and the procedures for maintaining information on the WWW.

A GIS is a special type of information system concerned with *geographically* referenced data. Specifically:

geographic information system

> A geographic information system is a computer-based information system that enables capture, modeling, storage, retrieval, sharing, manipulation, analysis, and presentation of geographically referenced data.

geospatial

In this book we use the term *geospatial* to mean "geographically referenced." Thus geospatial data is a special type of spatial data that relates to the surface of the Earth. The key components of a GIS are shown schematically in Figure 1.1. Several other terms and types of information system are closely related to GIS (see "GIS terminology" inset, page 3).

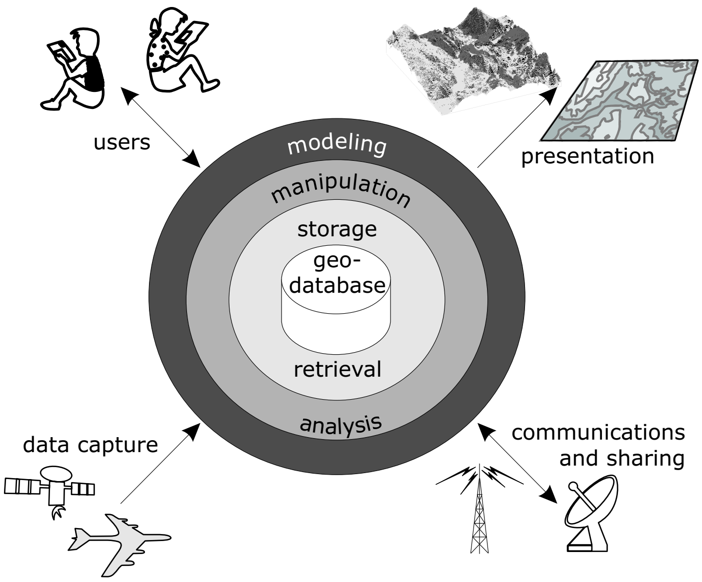


**Figure 1.1:** Schematic of a GIS

### 1.1.1 The shape of GIS

The world around us is both spatial and temporal, so we have a need for information that has spatial and temporal dimensions. Future decisions that affect us all, for example in planning new roads or cities, formulating

**GIS terminology** *There are several terms in common use that are more or less synonymous with GIS. A spatial information system (SIS) has the same functional components as a GIS, but may handle data that is referenced at a wider range of scales than simply geographic (for example, data on molecular configurations). A spatial database provides the database functionality for a spatial information system. A geographic database (or geodatabase) provides the database functionality for a GIS. An image database is fundamentally different from a spatial database or geodatabase in that the images have no structural inter-relationships or topological features. For example, a database of brain-scan images may be termed an image database. Computer-aided design (CAD) has some elements in common with GIS, and historically some GIS software packages have developed from CAD software. Unlike CAD, GIS software is used to manipulate data sets that are geographically referenced. As a consequence GIS software is usually used with larger data sets and more complex data models than CAD software. Conventionally, and in this book, the singular form of GIS may be used to refer to "the field of GIS," where no confusion is likely. While not synonymous with GIS, the terms geographic information science (GIScience) and geoinformatics are relevant to GIS as they are used to describe the systematic study of geographic information and geographic information systems.*

agricultural strategies, and locating mineral extraction sites, rely upon properly collected, managed, distributed, analyzed, and presented spatial and temporal information. A GIS may be thought of as a tool that is able to assist us with these tasks. This tool relies on several underlying elements. In the remainder of this section we look in more detail at these different elements, so describing the "shape" of GIS.

*Database element* At the heart of any GIS is the *database*. A database is a collection of data organized in such a way that a computer can efficiently store and retrieve the data. Databases are introduced in Chapter 2. An important element of a database is the *data model*. Some applications require relatively simple data models. For example, in a library system data about books, users, reservations, and loans is structured in a clear way. Many applications, including most GIS applications, demand more complex data models (consider for example a model of global climate). One of the main issues addressed in this book is the provision of facilities for handling these complex data models. Chapter 2 introduces some fundamental data modeling concepts, while Chapters 3 and 4 examine what makes geospatial data and geospatial data models special. database

In this book we discuss how complex geospatial data models can be represented in an information system. However, we do not discuss the suitability of particular data models for particular application areas: this is a topic for application domain experts. For example, a transportation geography textbook should offer insights into whether a particular model of transportation flow is appropriate for a particular application. We are only concerned with providing the general facilities to represent geospatial data models, which might include models of transportation flow.

# Fundamental spatial concepts

# 3

**Summary**

*Many different representations of space are commonly used within a GIS, and this chapter discusses the concepts underlying them. A familiar representation is **Euclidean geometry**, in which distances, angles, and coordinates may be defined. However, other representations are also important. **Sets** of **elements** provide a much simpler representation of space. **Topology** can be constructed based on the concept of a **neighborhood**. A **graph** comprises **nodes** connected by **edges**, and may be used to represent network spaces, such as a road network. A **metric space** formalizes the concept of distance between points in space.*

The term "space" is difficult to define (see "What is space?" inset, page 84). We all have an intuitive idea about the concrete space in which our bodies move. In the context of GIS, we normally use the term "space" to refer to "geographic space": the structure and properties of the relationships between locations at the Earth's surface. In this chapter we examine space more carefully by considering the different ways of representing and reasoning about geographic space.

A fundamental concept underlying these different representations is that of *geometry*. A geometry provides a formal representation of the abstract properties and structures within a space. Modern treatments of geometry are founded on the notion of *invariance*: geometries can be classified according to the group of transformations of space under which their propositions remain true. This idea was first proposed in 1872, by the German mathematician Felix Klein in his inaugural address to the University of Erlangen (known as the Erlangen Program).

geometry

invariance

To illustrate, consider a space of three-dimensions and our usual notion of distance between two points. Then a geometry is formed by the set of all transformations that preserve distances (that is, for which distance is an *invariant*). Into this set would fall translations and rotations,

**What is space?** *The notion of space is not easy to define. Gatrell (1991) defines space as "a relation defined on a set of objects," which includes just about any structured collection. This definition is too general to describe the spaces of interest in a GIS. A distinction is often made between the space we can apprehend using our visual perception (termed perceptual or small-scale space) and space that is too big for humans to observe all at once (termed transperceptual, large-scale space, or geographic space, see Kuipers, 1978; Mark and Frank, 1996; Montello, 1993). Zubin (1989) provides more detail and distinguishes four types of space: A-space contains manipulatable everyday objects, like phones and books; B-space contains objects larger than humans, but still observable from a single perspective, like buildings and buses; C-space contains geographic scenes that are too large to apprehend at one time, like landscapes; D-space contains objects that are too large for any human to truly experience, like the solar system or the galaxy. Freundschuh and Egenhofer (1997) give a full overview and synthesis of the different classifications of space that have been proposed. In GIS, we are primarily interested in geographic spaces, although there is also much interest in the relationship between the geographic spaces of GIS and other more general types of information space, such as the Internet and cyberspace (sometimes termed "cybergeography," see Dodge and Kitchin, 2002; Fabrikant and Buttenfield, 2001).*

because the distance between two points is the same before and after a translation or rotation is effected. Scalings (enlargements) would not be members because they usually change distance. Looking at this the other way round provides us with a definition of a geometry as the study of the invariants of a set of transformations. Thus the invariants of the set of translations, rotations, and scalings include angle and parallelism, but not distance.

Coordinatized Euclidean geometry provides a view of space that is intuitive, at least in Western culture, so this is our starting point (section 3.1). In later sections we discuss the most primitive space of all, just collections of objects with no other structure (section 3.2), and proceed to build up to richer geometries (sections 3.3–3.5). Finally, in section 3.6 we introduce fractal geometry, which is concerned with invariance in the scaling properties of objects. Because of the nature of the topic, the treatment will of necessity be sometimes abstract and formal, but examples are provided along the way. Some of the texts in the bibliography provide further background.

## 3.1 EUCLIDEAN SPACE

Euclidean space

Geospatial phenomena are commonly modeled as embedded in a coordinatized space, which enables measurements of distances and bearings between points according to the usual formulas (given below). This section describes this coordinatized model of space, called *Euclidean space*, which transforms spatial properties into properties of tuples of real numbers. We assume for simplicity a two-dimensional model, although all the concepts in this section can be generalized to higher dimensional spaces. For the Euclidean plane, we can set up a coordinate frame consisting of a fixed, distinguished point (*origin*) and a pair of orthogonal lines (*axes*), intersecting in the origin. origin axis

### 3.1.1 Point objects

A *point* in the plane of the axes has associated with it a unique pair of real numbers $(x, y)$ measuring its distance from the origin in the direction of each axis, respectively. The collection of all such points is the *Cartesian plane*, often written as $\mathbb{R}^2$. It is often useful to view Cartesian points $(x, y)$ as *vectors*, measured from the origin to the point $(x, y)$, having direction and magnitude and denoted by a directed line segment (see Figure 3.1). Thus they may be added, subtracted, and multiplied by scalars according to the rules: point Cartesian plane vector

$$(x_1, y_1) + (x_2, y_2) = (x_1 + x_2, y_1 + y_2)$$
$$(x_1, y_1) - (x_2, y_2) = (x_1 - x_2, y_1 - y_2)$$
$$k(x, y) = (kx, ky)$$

Given a point vector, $a = (x, y)$, we may form its *norm*, defined as follows: norm

$$\|a\| = \sqrt{(x^2 + y^2)}$$

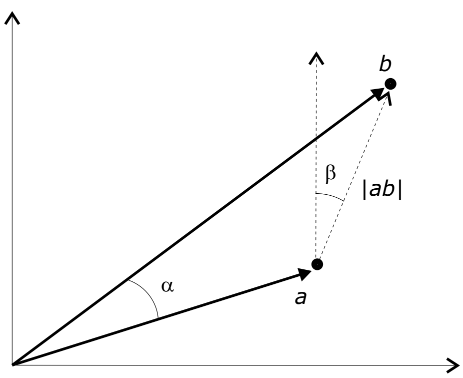


**Figure 3.1:** Distance, angle, and bearing between points (vectors) in the Euclidean plane

In a coordinatized system, measures of distance may be defined in a variety of ways (see section 3.5 on metric spaces). A *Euclidean plane* is a Cartesian plane with the particular measures of distance and angle given below. These measures form the foundation of most school geometry courses and refer to the "as the crow flies" concept of distance. Given points (vectors) $a, b$ in $\mathbb{R}^2$, the distance from $a$ to $b$, $|ab|$ (see Figure 3.1) is given by: Euclidean plane

$$|ab| = \|a - b\|$$

Suppose that the points $a, b$ in $\mathbb{R}^2$ have coordinates $(x_a, y_a)$ and $(x_b, y_b)$, respectively. Then the distance $|ab|$ is precisely the Pythagorean distance familiar from school days, given by:

$$|ab| = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}$$

# Structures and access methods

# 6

## Summary

*This chapter is about the organization of data in computer storage to facilitate efficient retrieval. The chapter begins by surveying some basic data structures and* ***index*** *methods for general-purpose databases, and then focuses of efficient spatial data retrieval. Some of the more important spatial data structures introduced include the* ***region quadtree****,* ***point quadtree****, and* ***2D-tree****. The chapter concludes with issues related to storing data referenced to the sphere rather than the plane.*

In this chapter there is a further movement from high-level conceptual models toward the machine level. In the last chapter, the emphasis was upon computationally appropriate ways of representing and manipulating different kinds of spatial data. We now move on to consider storage questions, and in particular storage structures that allow acceptable *performance*. In this chapter, the performance of an information system is taken to be measured in terms of database size and response times to queries. Other aspects of performance, such as the suitability and usability of the interface, are considered in Chapter 8. The discussion here begins with an introduction to the main issues of performance for general-purpose databases and then takes up these issues in the special context of geospatial data. We will see that while the fundamental principles of good database indexing practice still apply, spatial data access presents its own special problems.

performance

### 6.1 GENERAL DATABASE STRUCTURES AND ACCESS METHODS

The typical organization of a database is as a collection of *files*, each containing a collection of *records*, stored on a set of disks. Throughout this chapter we assume disks, either magnetic or optical, are being used as secondary storage media. We noted the main physical characteristics of a computer disk in Chapter 1. The atomic unit of data held on a disk is referred to as the disk *block*. The time taken to transfer a disk block to or from a disk has three components:

block

- the time taken for the mechanical movement of the disk heads across the disk to the correct track, termed *seek time*;
- the time taken for the disk to rotate to the correct position, termed *latency*; and
- the time required to transfer the block into the CPU, *CPU transfer time*.

latency

CPU transfer time

As noted in Chapter 1, seek time is the dominant factor determining performance in data retrieval from disks. Data structures in secondary storage that lessen the mechanical movement of the disk heads will therefore result in improved database performance. To minimize disk head movement, data is ideally placed in such a way that blocks which are often accessed together are close together on the disk. Lessening disk head movement will also be a consequence of the construction of appropriate indexes, so that unnecessary searches, maybe of entire files, are avoided.

Databases, by their very nature, are usually designed to be flexible and to respond well to a wide variety of queries. In general, queries are unknown in advance, although where predictable query patterns do exist they should be identified during the system development process. Another difficulty is caused if the database is highly dynamic, with its content changing considerably in the course of its lifetime. Therefore, the physical placement of data on the disk can only provide a partial solution to performance issues. Efficient indexes and access methods are also needed to achieve good performance.

### 6.1.1 File organization and access methods

Before getting to the main discussion, we introduce the fundamental concepts of file, record, and field.

- A (named) place for a data item in a record is termed a *field*. Note that this sense of "field" is unrelated to "field" in the sense of field-based models.
- A sequence of fields related to a single logical entity is termed a *record*.
- A sequence of records, usually all of the same type, is organized into a *file*. A field that serves to identify each record within a file unambiguously is called a *key field*.

field

record

file

key field

The terms field, record, and file in data organization are clearly analogous to attribute, tuple, and relation in database terminology. In file organization terms, a database is a collection of related files. Figure 6.1 illustrates the organization of fields, records, and files within an example weather database. Fields, such as the temperature field for Bangor, provide a place for data items. Individual data items, such as ‘82°F’, are stored together in records, such as the weather report record for Bangor, Maine, on July 7 2003. Sequences of related records, such as the weather for Bangor for 2003, are organized into files. Collections of related files, such as the weather report for all cities in Maine, form a database.

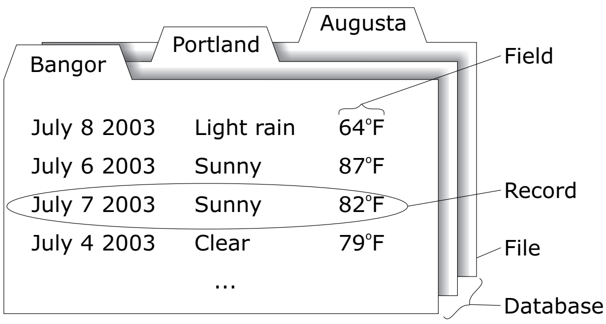


**Figure 6.1:** Data organization into field, record, file, and database

Files are physically placed upon a disk by assigning disk blocks to hold records. If the disk block is smaller than the record size, then records will be spread across blocks. Otherwise (and more usually) each block contains several records. The term *file organization* is used to refer to the physical organization of the records on the secondary storage, the manner in which blocks of records are linked, and the way new records are inserted into storage.

file organization

### 6.1.2 Unordered files and linear searches

In an *unordered* file organization, new records are inserted into the file in the next physical position on the disk, either in the last used disk block or in a new disk block. Insertion of a new record is therefore very efficient. However, the unordered file has no structure beyond entry order, so retrievals will require a search through each record of the file in sequence. If all that is required is that the file is accessed sequentially, for example to print out lists of weather reports for a particular location, then an unordered file organization is acceptable. However, problems arise with direct access, where specific information is required from targeted records. For example, retrieving those days on which the temperature rose above 80°F requires that the value of the temperature field of each record be examined in turn. Such a search is called a *linear search*, or a *brute-force* approach, and is to be avoided if possible. For $n$ records a linear search may need to retrieve every record. A linear search therefore has linear time complexity, $O(n)$. We will see shortly how indexes can help to solve this problem.

unordered file organization

linear search

Another difficulty arises for a highly dynamic file, where as records are deleted “holes” are created. It is possible to modify unordered file

# Interfaces 8

## Summary

*The field of **human-computer interaction** (HCI) is concerned with the design, evaluation, and implementation of effective **user interfaces** between humans and computing devices. A good interface should be both **intuitive** (easy to learn and use) and **expressive** (able to specify and perform the desired tasks efficiently). GIS interfaces are often based on the **map metaphor**: they exhibit similar characteristics to conventional paper maps. However, GIS interfaces can extend the map metaphor in several ways, for example using animated, three-dimensional, non-visual, and multimodal displays.*

The ability of computer-based systems to interoperate is an important feature of any GIS architecture, covered in the previous chapter. However, the success of any computer-based system ultimately rests on whether it can be used effectively by *people*. Just as GIS architectures are designed to promote interoperability between different computers, so GIS interfaces are designed to ensure that GISs and people can "interoperate." In this chapter, the essential characteristics of GIS interfaces are explored. Section 8.1 introduces the basic principles of computer interfaces. The main interface styles that may be used in a GIS are then explored in more detail, starting with conventional map-based interfaces in section 8.2 and moving on to the roles of animation, three-dimensional displays, non-visual interfaces, and feedback in section 8.3. Section 8.4 addresses the development in GIS interfaces for different types of user tasks.

### 8.1 HUMAN-COMPUTER INTERACTION

human-computer interaction

The term "human-computer interaction" (HCI) was first used during the 1980s to describe the interaction between computer systems and people. The term is also used to refer to the field of HCI: the study of the design, evaluation, and implementation of the interfaces between computing devices and people. Figure 8.1 summarizes the key components of HCI: the human, the computer, and the interaction between them. This section explores in more detail how information is exchanged between the computer and the human, and what types of interface are used to facilitate interaction.

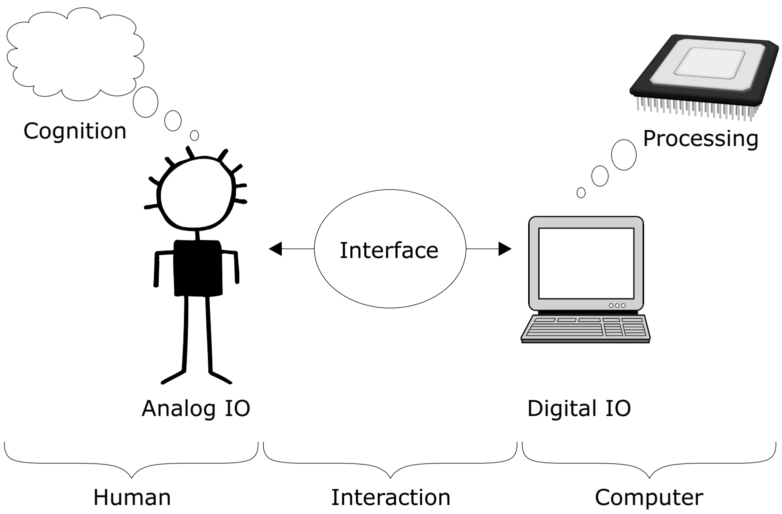


**Figure 8.1:** Components of human-computer interaction

### 8.1.1 Input-output channels

IO channel

Humans and computers are able to send and receive information in several different modes, termed *input-output* (IO) channels. Computers send information on IO channels that are detectable by human senses (see "Human senses" inset, on the next page). Computers are equipped with devices to detect information sent by human users on similar IO channels. For example, a PC receives information from a human user primarily on the *haptic* IO channel ("haptic" means related to touch), via the keyboard and the mouse. In return, information is sent from the PC to its human user on the visual IO channel, via a VDU (visual display unit) or monitor.

haptic

The same IO channel can be used for sending information in both directions, from computer to human and from human to computer. For example, auditory information can be sent via a computer speaker and received by human hearing, or sent by the human voice and received by a computer microphone. To avoid confusion, the term *display* is often used to refer to output from a computer to a human for any IO channel, while *input* is normally reserved for input to a computer from a human. Information displayed on more that one IO channel at a time may be useful in reinforcing important messages. For example, to alert a PC user to a warning, a visible message will often be accompanied by an audible "bleep." Systems that enable display or input of information on more than one IO channel (mode) at a time are termed *multimodal*.

display

input

multimodal

# 10 Time

**Summary**

*A* ***spatiotemporal information system*** *extends a GIS by storing and managing spatial and temporal information.* ***Snapshot*** *representations of spatiotemporal phenomena represent the state of the geographic application domain at a particular time. Object* ***lifelines*** *extend the snapshot representation by explicitly storing information about changes. More recent advances in spatiotemporal databases have focused on the explicit representation of spatiotemporal entities, such as* ***events*** *and* ***processes****. This chapter examines some of the basic concepts involved in introducing time into a GIS, ranging from the underlying data model to spatiotemporal data structures to facilitate data retrieval.*

We have moved from a data-poor to a data-rich information society, and much of this data has both spatial and temporal components. Geographic information systems are now beginning to have some temporal functionality, and a *spatiotemporal information system* manages information that is both (geo)spatially and temporally referenced. While truly spatiotemporal information systems are still primarily a research topic, GISs are beginning to be extended so that they can offer some practical temporal functionality. A wide range of further spatiotemporal functionality is now at the margins of practical application. There is a variety of ways in which such functionality can be integrated. This chapter details some of the issues facing such spatiotemporal systems.

spatiotemporal information system

There are many potential application domains for spatiotemporal systems, including environmental change, transportation, socioeconomic and demographic applications, health and epidemiology, multimedia, governance and administration, and defense. In addition to these more traditional spatiotemporal application areas, the increased use of real-time, mobile and *in situ* sensors is leading to many new potential applications for spatiotemporal data models and systems. Many of these applications relate to mobile location-aware and pervasive systems, introduced in Chapter 7).

## 10.1 INTRODUCTION: "A BRIEF HISTORY OF TIME"

At a fundamental level, entities in the world may be divided into happenings; objects and their properties; roles and relationships; and a few basic notions like location (in time and space). When we ask a question of an information system, we are generally interested in knowing something about these entities. Traditional databases manage information about objects, along with their properties and relationships, as in, for example, the entity-relationship model (Chapter 2). Adding temporality allows the possibility of managing information about the states of objects at particular times and places, but also the *properties of what happened, when, how, and why*. Spatiotemporal database research is at present primarily focused on answering questions about objects of the form "what–where–when" (for example, "Who owned the land at location $l$ and at time $t$?"). Research into next generation systems is concerned with answering questions about what happened, where, when, how, and why. To achieve this, functionality for the analysis of relationships between these happenings, such as causality, is needed.

Geographic phenomena have both spatial and temporal components. Figure 10.1 illustrates both natural and artificial spatiotemporal geographic phenomena. Figure 10.1a shows a satellite image of a wildfire in Arizona. Smoke from the burning vegetation can be seen in the center of the image, distinct from atmospheric cloud (bottom left of image). Figure 10.1b shows urban traffic at an intersection in the UK. The images are themselves static, but the underlying phenomena are of course dynamic.


a. Wildfire in Tucson, Arizona, June 2003 (NASA ASTER image)

b. Urban traffic flow in the UK (www.freeimages.co.uk)

**Figure 10.1:** Examples of natural and artificial spatiotemporal phenomena

Dynamic geographic entities are characterized not only by spatial and attribute components, but also by temporal references. Geographic phenomena, such as events, actions, and processes, are explicitly temporal.

A spatiotemporal information system must manage data about all these time-varying real world entities. Efforts to incorporate temporal aspects of geographic phenomena into data models and information systems are continuing, but are by no means complete. Consider the following stages in a "brief history of time."

### 10.1.1 Stage zero: Static representations

Traditional spatial and geographic information systems hold only a single state of the "real world." This state is almost always the most recent in time for which the data was captured. Interactions with the system are "timeless," in that only information contained in the single state can be retrieved. Most of the systems and models we have met up to now in this book have been concerned with static representations.

### 10.1.2 Stage one: The snapshot metaphor

Stage one uses the *snapshot* metaphor to show dynamic phenomena as a collection of *timestamped* states. Much research into spatiotemporal information systems has focused on the notion of sequences of snapshots. Indeed, snapshots can be a powerful mechanism for understanding change. For example, Figure 10.2 shows the growth of the University of Maine campus and the surrounding region from 1902 to 1955 as a series of three temporal snapshots. snapshot

Underlying such stage one representations are models of time. For example, if time is viewed as a linear dimension, shown on a timeline (i.e. timestamps are linearly ordered), then the snapshot collection becomes a sequence. Models for time, including linear, branching, and cyclical structures, as well as distinctions between dense and discrete frameworks, are discussed later in the chapter.

### 10.1.3 Stage two: Object lifelines

The difficulty with stage one is that the static nature of the individual timestamped states dictates the generally static nature of the representation. Thus, it is a problem in stage one representations to identify dynamic phenomena, such as birth, change, and death. Figure 10.3 shows map detail from another part of the region around the University of Maine. The figure shows that an airport was constructed between 1902 and 1946, but only implicitly through comparison of the two states. The snapshot metaphor offers no mechanism for *explicitly* representing information about the time or the occurrence of events such as the construction of an airport.

Stage two begins to address these issues. Object *lifelines* are designed to explicitly represent changes of state in a single object and interactions between different objects. Figure 10.4 shows the consequences of interaction between objects. The events creation, transmission, reappearance, disappearance, transformation, cloning, and deletion are all explicitly lifeline