6 Graphics Part Hierarchies

6.1 Introduction

When we look at the following picture, a schematic illustration of the front panel of a car radio, we intuitively understand that it is made up of parts: the radio has two switchboards, each switchboard consists of four buttons, a button consists of a square and a circle.

![Front panel of a car radio. With this drawing a part hierarchy is intuitively associated: the radio has two switchboards, each switchboard consists of four buttons, a button consists of a square and a circle.](image)

The organization of items in the form of part hierarchies is one of the most frequently used methods of modeling and describing phenomena in the real world. The idea behind this is that of constructing or describing complex objects through composition of simpler ones. It will reduce the overall complexity of a problem. Such organization of objects allows us to describe the construction of a whole from parts. Beyond this, the principle of re-using similar parts, i.e., parts with the same geometrical description, is often expressed in the context of part hierarchies. Part hierarchies play an important role in many areas of science and technology.

In computer graphics, part hierarchies are of interest wherever complex graphics representations can be interpreted as conglomerations of simpler parts—which is the case in most applications of graphics. An interactive graphics editor, such as MacDraw, is a typical application program based by the concept of composition: simple graphics primitives may be selected from a menu and grouped to form higher aggregates. Copy-operations serve to multiply parts and support the idea of the repeated use of objects having the same building pattern.

Part hierarchies are important in other fields of computer science, too—for instance, artificial intelligence. They are often used there in modeling and repre-
senting so-called world knowledge. We must note that a part hierarchy differs from a class hierarchy as understood by object-oriented programming. A class hierarchy expresses a behavior-relation, whereas a part hierarchy expresses a composition-relation. In the field of computer graphics, the word inheritance is sometimes used to express semantic aspects in connection with the part hierarchy. For example, one speaks of parts inheriting attributes from the structure they belong to. So the automatic transfer of characteristics of the whole to its constituents is called inheritance here. This use of the word inheritance is not to be confused with inheritance in the sense of creating subclasses. For this reason we will use the term inheritance only for latter sense, in the context of a part hierarchy we will speaking of transferring attributes down to parts of a hierarchy.

6.2 Part Hierarchies and Computer Graphics

In all those application areas of graphics which deal with structures that can be described naturally as part hierarchies, a graphics system that optimally supports the semantic aspects of a part hierarchy is highly desirable. Consequently that such a system should provide for the representation of knowledge about the composition of a part hierarchy. In addition, it should offer the basic functions for their manipulation.

6.2.1 Requirements

The semantics of a part hierarchy are described by the operations that can be applied to the hierarchy. To a certain extent, their scope can be chosen freely. There exists, however, a hard kernel of operations without which it would be senseless to speak of a part hierarchy. If an interactive graphics system claims to support part hierarchies effectively, then we expect support in the following areas of functionality:

- Construction of a part hierarchy.
- Editing the hierarchy.
- Assignment and inquiring of attributes.
- Graphics input operations for the identification of parts of the hierarchy.

6.2.2 Criteria for Ideal System Support

Important criteria for the quality of a graphics system that supports a part hierarchy are naturalness and simplicity.

- Naturalness: the functional description of such a system by a programming language should be as close as possible to the semantics of the part hierarchy. Therefore, the graphics system must offer constructs that are intuitively related to the description of the part hierarchy.
- Simplicity: all frequently used operations for the manipulation of a part hierarchy should be supported directly. It should be possible to formulate each of these operations as a single expression, i.e., as an atomic action.