

# What is Fusion?

Fusion is a systematic software development method for object-oriented software development. Developed at Hewlett-Packard Laboratories in Bristol, England, the method integrates and extends the best features of earlier object-oriented methods. Fusion is a full-coverage method, providing a direct route from a requirements definition through analysis and design to a programming language implementation.

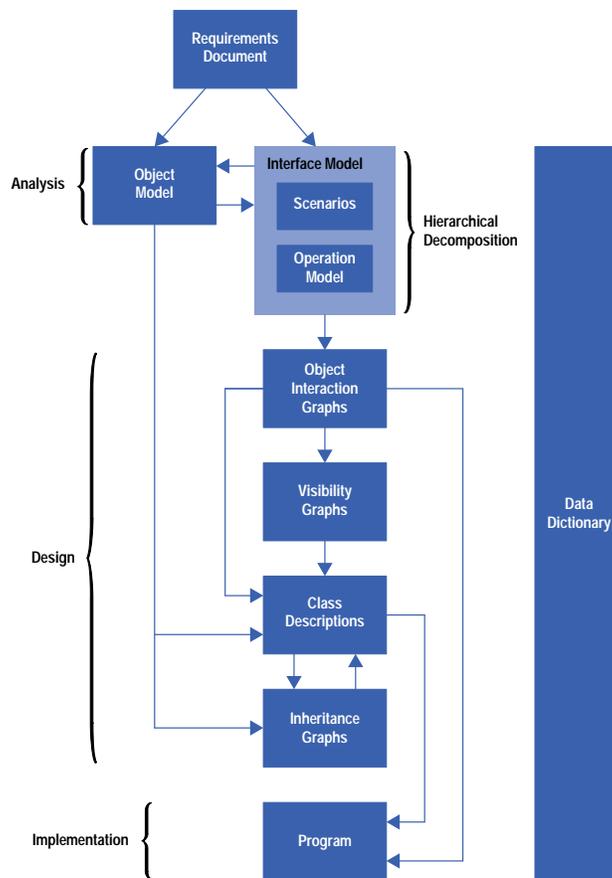
## What Fusion Offers

- It provides a process for software development. It divides this process into phases and says what should be done in each phase. It gives guidance on the order in which things should be done within phases so that the developer knows how to make progress. It provides criteria that tell the developer when to move on to the next phase.
- It provides a comprehensive, simple, well-defined notation for all of its models. Because this notation is based on existing practices, it is easy to learn.
- It provides management tools for software development. The outputs of the different phases are clearly identified, and there are cross-checks to ensure consistency within and between phases. Each phase has its own techniques and addresses different aspects of translating a requirements document into executable code.
- It is adaptable. A lightweight version can be used in projects that cannot afford the effort required to use the full version, or parts of the process or notation can be used within other development processes to address their weak points.

## The Process

The Fusion method structures the development process into analysis, design, and implementation phases (see Fig. 1).

*Fig. 1. The Fusion process.*



## Analysis

During the analysis phase the analyst defines the intended behavior of the system. Models of the system are produced, which describe:

- The classes of objects that exist in the system
- The relationships between those classes

- The operations that can be performed on the system
- The allowable sequences of those operations.

## Design

The designer chooses how the system operations are to be implemented by the run-time behavior of interacting objects. Different ways of breaking an operation into interactions can be tried. During this process, operations are attached to classes. The designer also chooses how objects refer to each other and what the appropriate inheritance relationships are between classes.

The design phase delivers models that show:

- How operations on the system are implemented by interacting objects
- How classes refer one to another and how they are related by inheritance
- The attributes of and operations on classes.

Designers may need to investigate the substructure of some classes and their operations in more detail. They do so by applying the analysis and design techniques to those classes, regarding them as a subsystem.

## Implementation

The implementer must turn the design into code in a particular programming language. Fusion gives guidance on how this is done in the following ways:

- Inheritance, reference, and class attributes are implemented in programming-language classes.
- Object interactions are encoded as methods belonging to a selected class.
- The permitted sequences of operations are recognized by state machines.

Fusion also maintains a data dictionary, a place where the different entities of the system can be named and described. The data dictionary is referenced throughout the development process.

In summary, Fusion is a complete, yet lightweight development method that can be tailored to meet the different needs of software projects.

---

---

Derek Coleman  
Professor of Computer Science  
University of London

- 
- 
- ▶ [Return to Article 3](#)
  - ▶ [Go to Article 4](#)
  - ▶ [Go to Table of Contents](#)
  - ▶ [Go to HP Journal Home Page](#)