Implementation and integration

In this chapter we will:

- Realize that the implementation and integration phases must be performed in parallel
- Discover the advantages and disadvantages of various techniques for the implementation and integration phase
- Learn about the distinct types of testing that are carried out during the implementation and integration phase

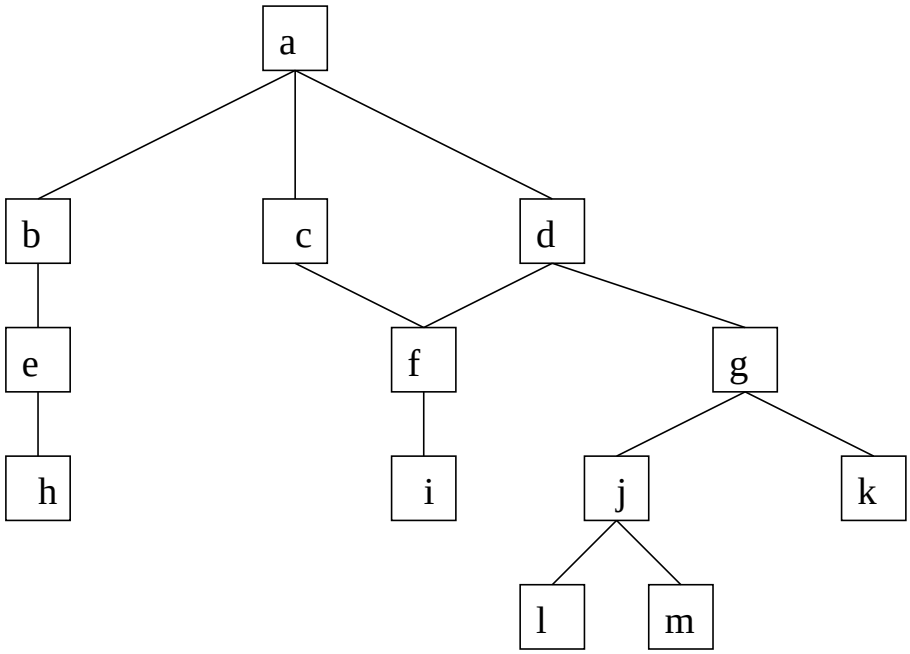

Figure 10.1 Structure chart of a product.

If implementation and integration are performed sequentially then

First, Each function is implemented and tested on its own

Second, all functions are linked together and tested.

Suppose the result crashes. Where is the fault?

Two problems of this scenario:

- It has poor error localization possibilities,
- It is hard to develop code and test it because it requires additional effort: implementation of test drivers and function stubs.

Top-down implementation and integration

Gradually implement and test beginning at root and replacing stubs with functions.

Independent paths can be implemented and integrated separately.

How about error localization? We integrate a new function into a tested one (by replacing its stub). Suppose, that an error occurs. This is how we try to localize it:

- Error is in the new function
- if not, then it is in the interface
- if not, then it is in the old function.

Additional benefit is early detection of design faults. Upper-level functions in the structure chart tend to be of logical nature rather than operational. If we discover and fix a logical error early, operations (lower blocks) must not be rewritten or discarded as it could occur with the sequential scenario.

Drawback of this scenario is that lower-level functions are tested less frequently than upper-level ones.

This is especially critical if lower-level functions are going to be included into a library of reusable functions.

Question: Why?

Hint: Lower-level functions are tested only via upper-level functions.

Bottom-up implementation and integration

Advantages:

- Good fault localization
- Thoroughly tested functions

Disadvantages: Design errors detected late.

To have the best of both scenarios: *sandwich implementation.*

Question: What is

- Top
- Bottom
- Filling

of the sandwich?

Product and acceptance testing

Product test is run by the SQA team and consists of

- Black-box test: does the product satisfy the specification?
- Robustness testing: how does the product react to stress situations or incorrect input?
- Check the constraints
- Check the documentation.

Customer performs acceptance test. Now the above points are run on actual data, no test data!

Maintenance phase

Types of maintenance

- Corrective maintenance
- Adaptive maintenance
- Perfective maintenance
- Enhancement

Question: Why is maintenance difficult?

Maintenance programmer must have

- Good skills in error localization, bug fix and testing
- Proper tools (e.g. version control).