東北大學
Northeastern University

软件工程

张爽

东北大学软件学院

Make an analogy with software and hardware

What is computer hardware composed of ?

What is a Module?

- **What is a module?**

  A lexically contiguous sequence of program statements, bounded by boundary elements, with an aggregate identifier.

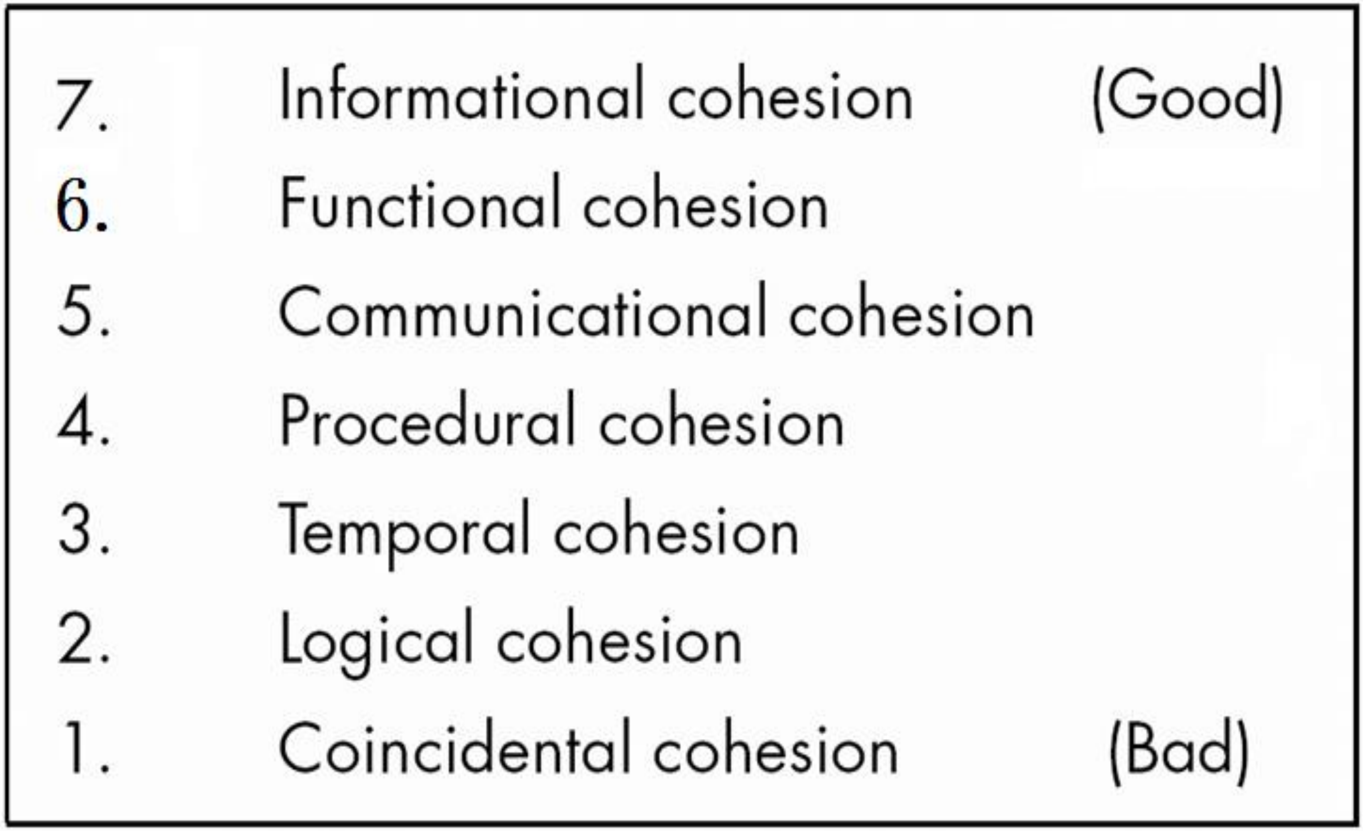
Module

- Method for breaking up a product into modules
  - Maximal interaction within module, and
  - Minimal interaction between modules
- Module cohesion
  - Degree of interaction within a module
- Module coupling
  - Degree of interaction between modules

4.1 Cohesion

Cohesion

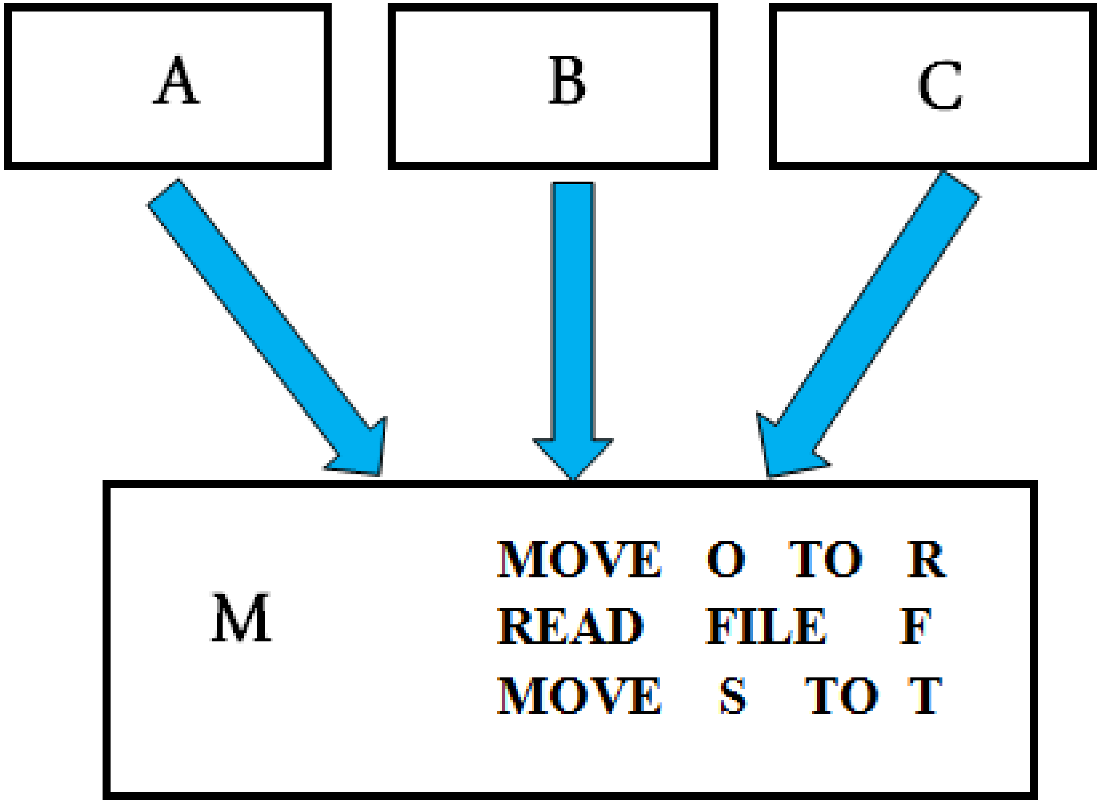
- Degree of interaction within a module
- Seven categories or levels of cohesion



7.	Informational cohesion	(Good)
6.	Functional cohesion	
5.	Communicational cohesion	
4.	Procedural cohesion	
3.	Temporal cohesion	
2.	Logical cohesion	
1.	Coincidental cohesion	(Bad)

1. Coincidental Cohesion

- **A module has coincidental cohesion if it performs multiple, completely unrelated actions.**

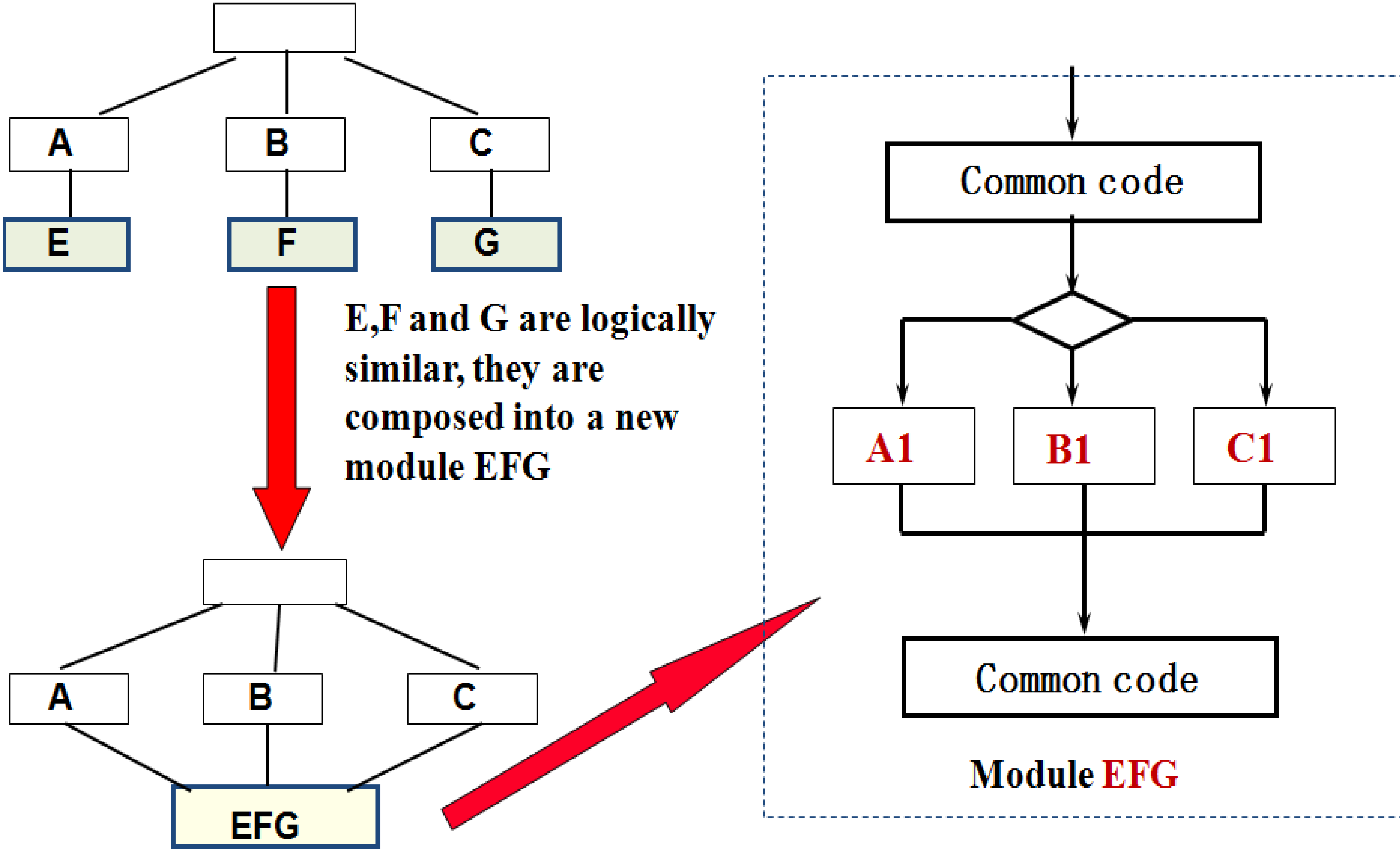


Why Is Coincidental Cohesion So Bad?

- Degrades maintainability
- Modules are not reusable
- This is easy to fix
  - – Break into separate modules each performing one task

2. Logical Cohesion

- **A module has logical cohesion when it performs a series of related actions, one of which is selected by the calling module.**



E,F and G are logically similar, they are composed into a new module EFG

Why Is Logical Cohesion So Bad?

- The interface is difficult to understand.
- Difficult to modify
- Code for more than one action may be intertwined.
- Increase coupling
- Low efficiency

3. Temporal Cohesion

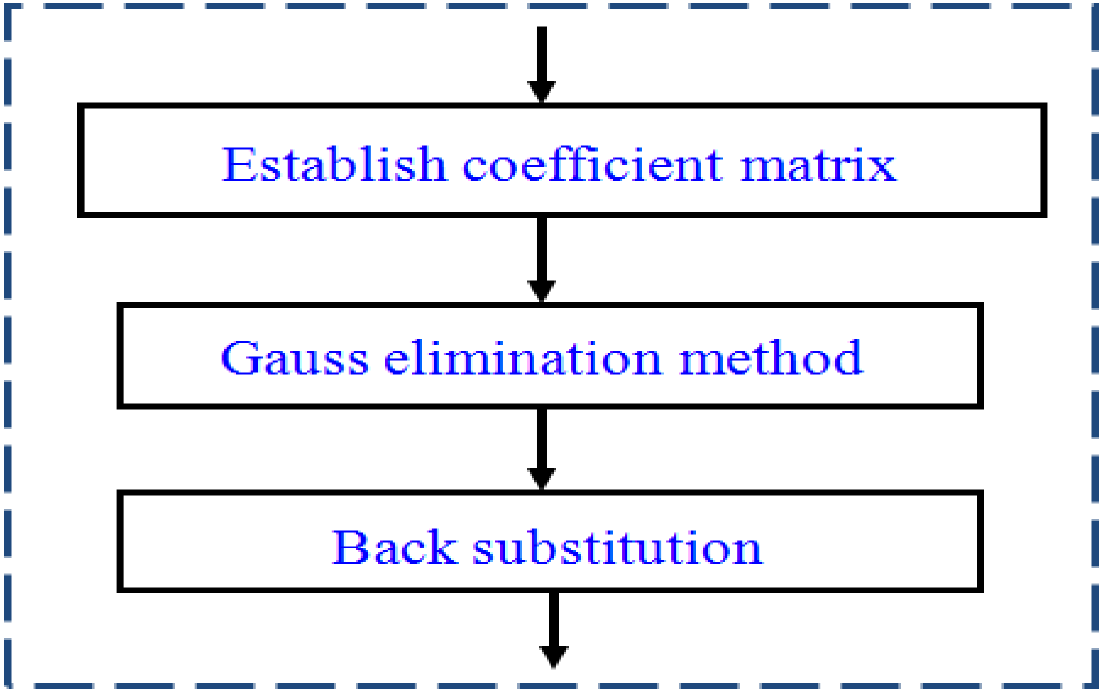
- A module has temporal cohesion when it performs a series of actions related in time
- The actions in the module must execute in the same time.
- Example:
  - Initialization module
  - Error handling module
  - System termination module

Why Is Temporal Cohesion So Bad?

- Actions of this module are weakly related to one another, but strongly related to actions in other modules.
- Not reusable

4. Procedural Cohesion

- **A module has procedural cohesion if it performs a series of actions related by the procedure to be followed by the product.**



Establish coefficient matrix

Gauss elimination method

Back substitution

Gauss elimination algorithm

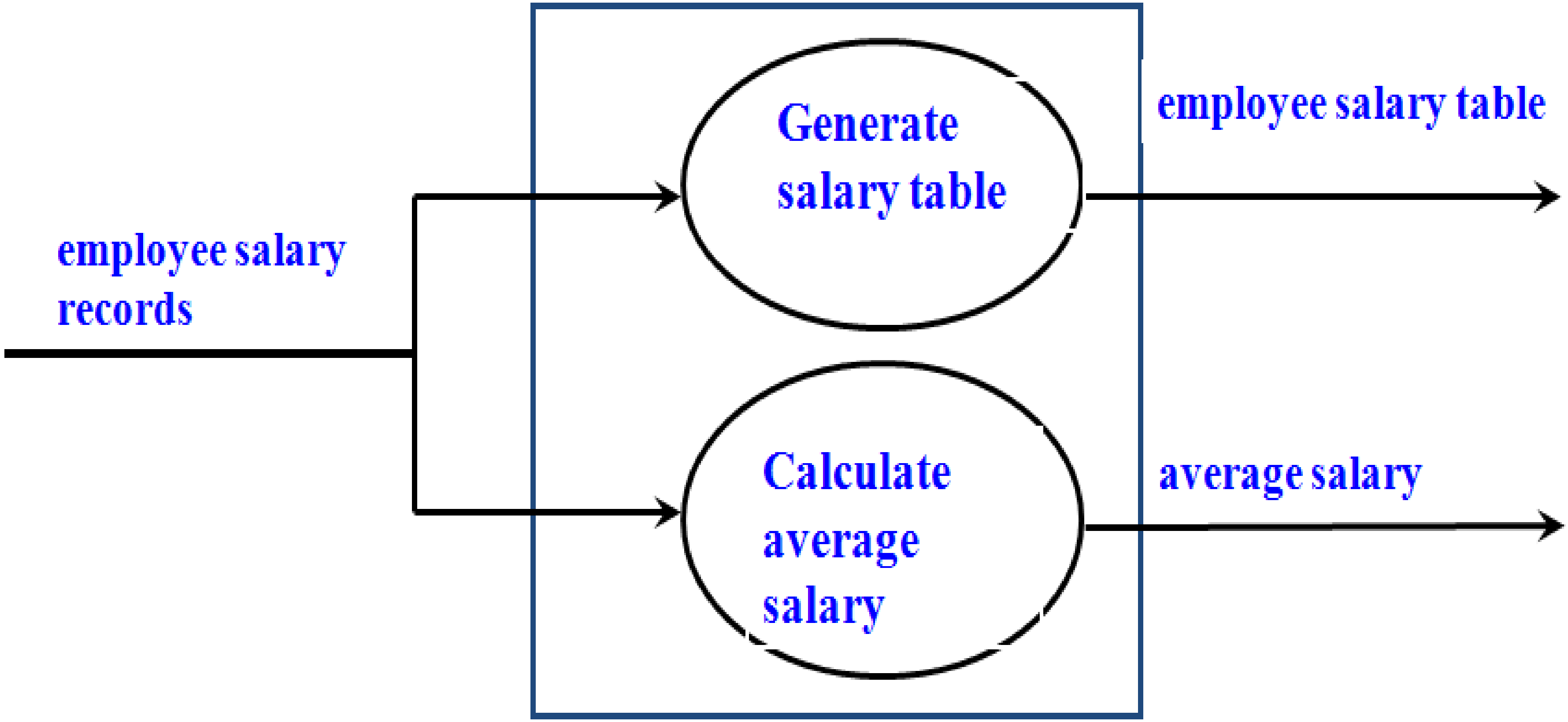
Why Is Procedural Cohesion So Bad?

- **Actions are still weakly connected, so module is not reusable.**

5. Communicational Cohesion

- A module has communicational cohesion if it performs a series of actions related by the procedure to be followed by the product, but in addition all the actions operate on the same input or output data.

5. Communicational Cohesion



Why Is Communicational Cohesion So Bad?

- Still lack of reusability

6. Functional Cohesion

- **Module with functional cohesion performs exactly one action.**

Why is functional cohesion so good?

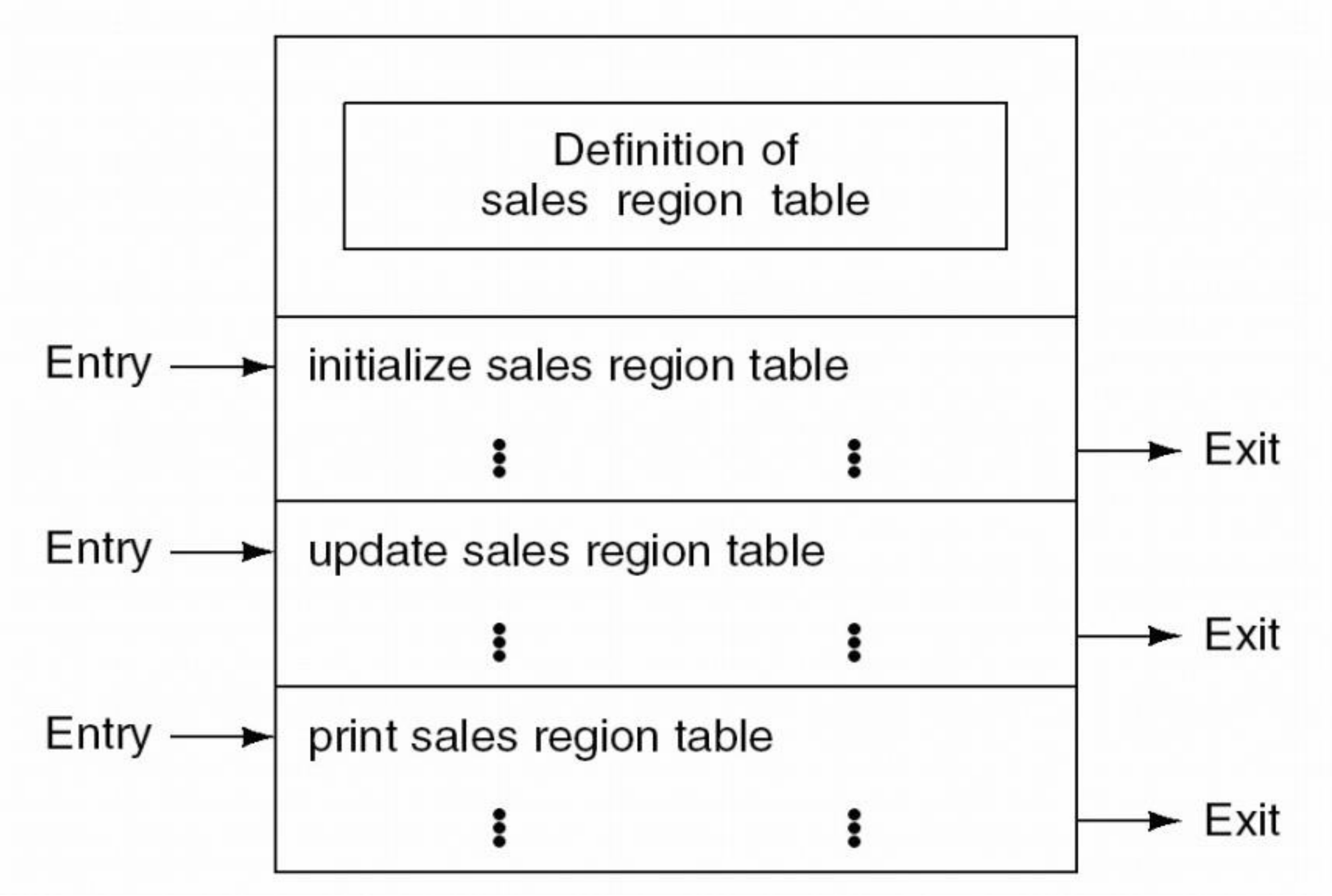
- ➢ More reusable
- ➢ Corrective maintenance easier
    - Fault isolation
    - Fewer regression faults
- ➢ Easier to extend product

7. Informational Cohesion

- A module has informational cohesion if it performs a number of actions, each with its own entry point, with independent code for each action, all performed on the same data structure.

Why Is Informational Cohesion So Good?

- **Essentially, this is an abstract data type**



Definition of sales region table
Entry
initialize sales region table
Exit
Entry
update sales region table
Exit
Entry
print sales region table
Exit