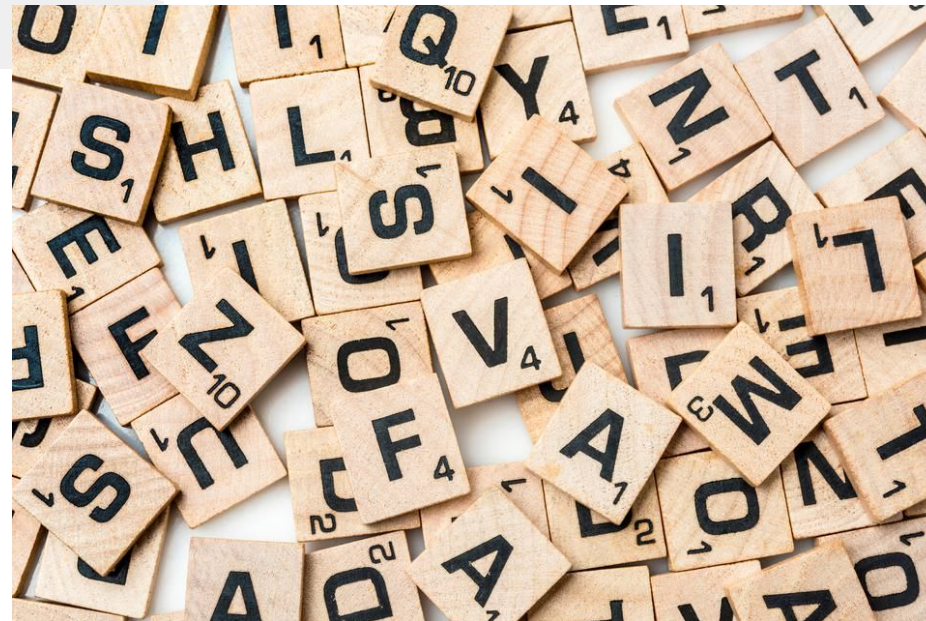


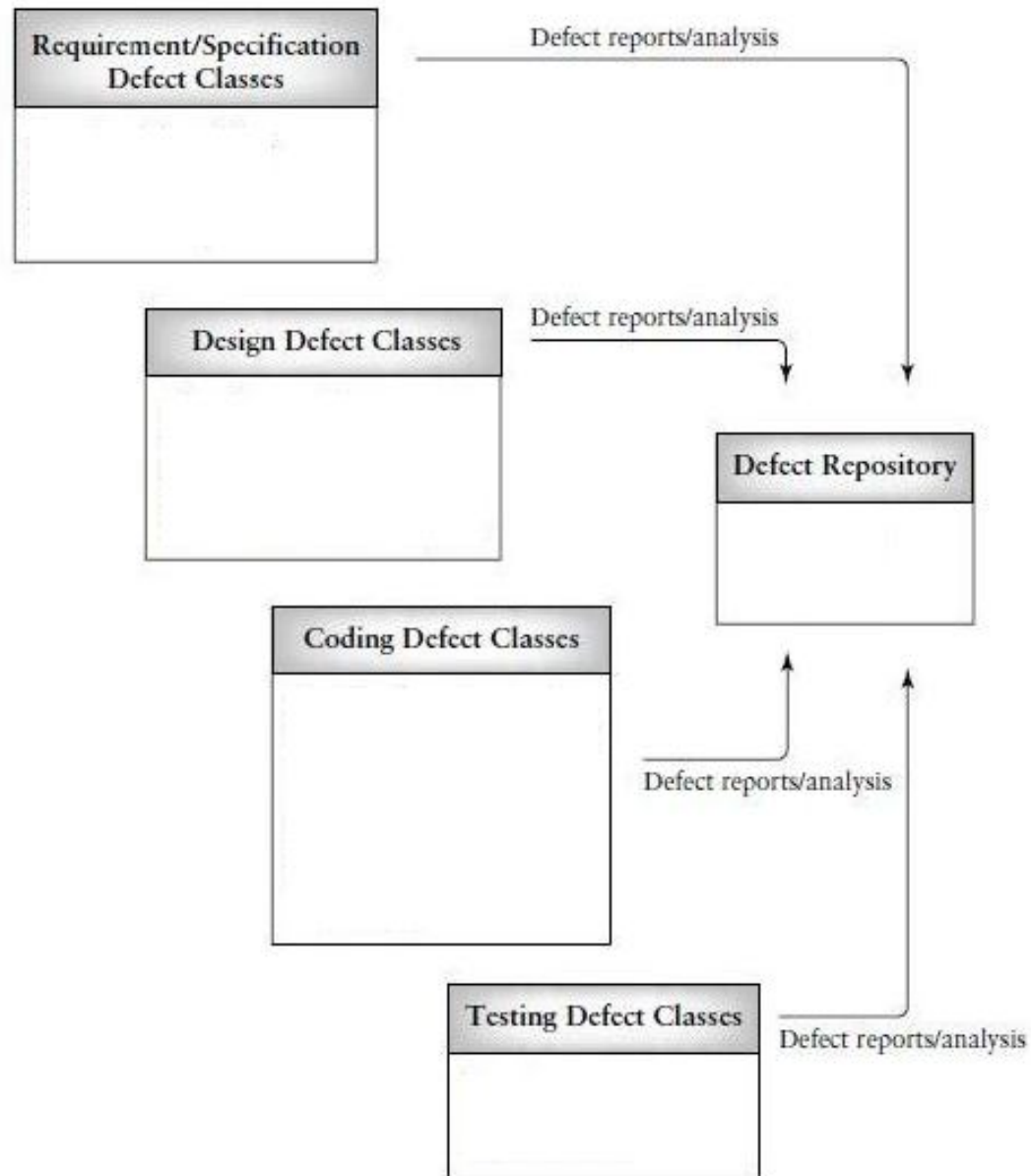


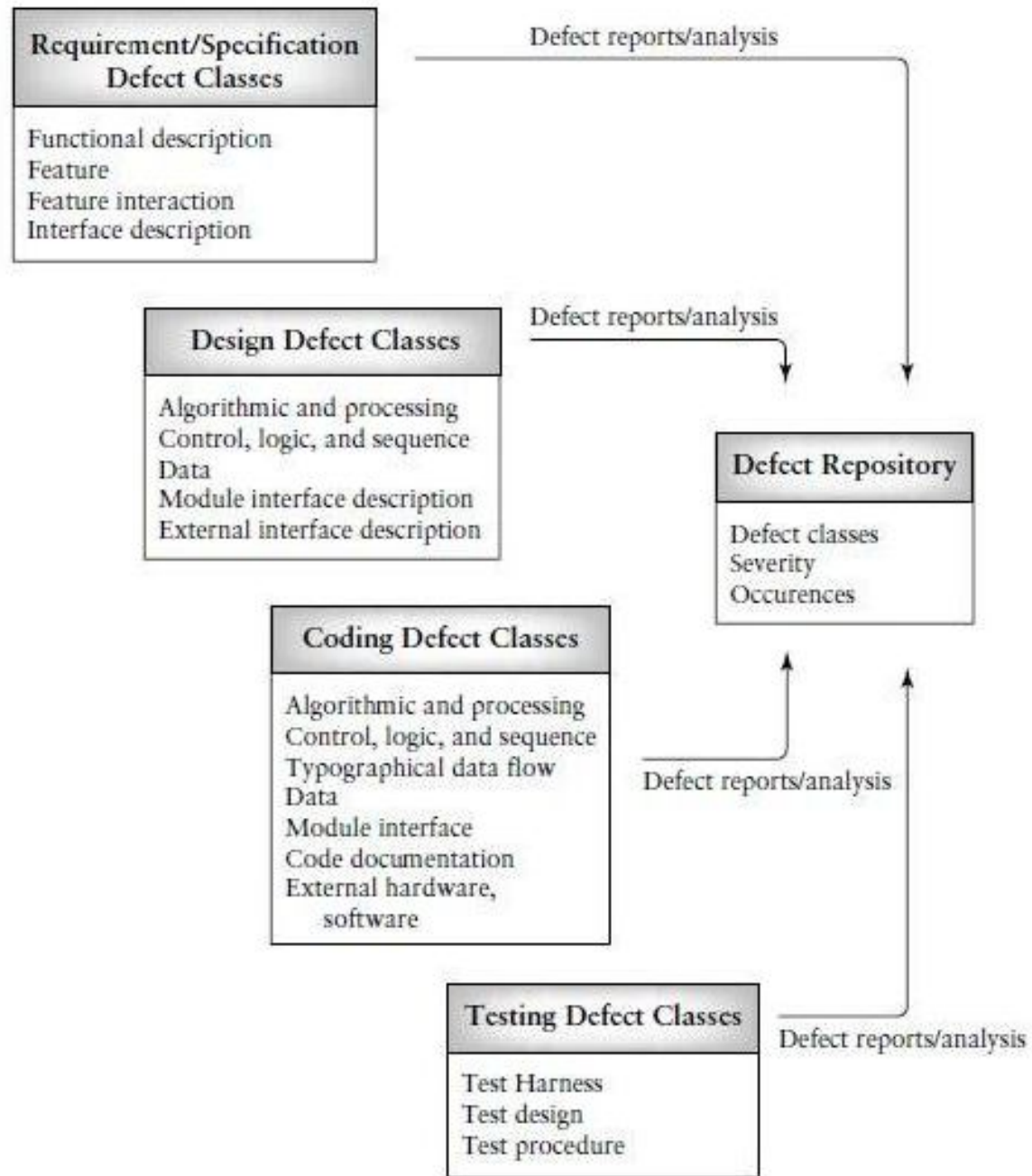
Defect Classes



Activity









Requirement and Specification defects

▶ **Functional Description Defects**

- ▶ overall description of what the product does, and how it should behave (inputs/outputs), is incorrect, ambiguous, and/or incomplete.



▶ **Feature Defects**

- ▶ Distinguishing characteristics of a software component or system.
- ▶ Functional and Quality requirements.
- ▶ Feature descriptions - missing, incorrect, incomplete, or superfluous.

▶ **Feature Interaction Defects**

- ▶ Incorrect description of how the features should interact.

▶ **Interface Description Defects**

- ▶ How the target software is to interface with external software, hardware, and users.



Design defects

- ▶ **Algorithmic and Processing Defects**
 - ▶ Processing steps in the algorithm as described by the pseudo code are incorrect.
- ▶ **Control, Logic, and Sequence Defects**
 - ▶ Logic flow in the pseudo code is not correct.
- ▶ **Data Defects**
 - ▶ Incorrect design of data structures.
- ▶ **Module Interface Description Defects**
 - ▶ Incorrect, and/or inconsistent parameter types, an incorrect number of parameters, or an incorrect ordering of parameters.
- ▶ **Functional Description Defects**
 - ▶ Incorrect, missing, and/or unclear design elements
- ▶ **External Interface Description Defects**
 - ▶ Incorrect design descriptions for interfaces with COTS components, external software systems, databases, and hardware devices (e.g., I/O devices).



Coding defects

```
for i in people.data.users:
    response = client.api.statuses.user_timeline.get(screen_name=i.scre
    print 'Got', len(response.data), 'tweets from', i.screen_name
    if len(response.data) != 0:
        ldate = response.data[0]['created_at']
        ldate2 = datetime.strptime(ldate, '%a %b %d %H:%M:%S +0000 %Y')
        today = datetime.now()
        howLong = (today-ldate2).days
        if howLong < daywindow:
            print i.screen_name, 'has tweeted in the past', daywindow,
            totaltweets += len(response.data)
            for j in response.data:
                if j.entities.urls:
                    for k in j.entities.urls:
                        newurl = k['expanded_url']
                        urlset.add((newurl, j.user.screen_name))
        else:
            print i.screen_name, 'has not tweeted in the past', daywind
```

▶ **Algorithmic and Processing Defects**

- ▶ Unchecked overflow and underflow conditions, comparing inappropriate data types, converting one data type to another, incorrect ordering of arithmetic operators (perhaps due to misunderstanding of the precedence of operators), misuse or omission of parentheses, precision loss, and incorrect use of signs.

▶ **Control, Logic and Sequence Defects**

- ▶ Incorrect expression of case statements, incorrect iteration of loops (loop boundary problems), and missing paths.

▶ **Typographical Defects** - incorrect spelling of a variable name

▶ **Initialization Defects** - initialization statements are omitted or are incorrect.

▶ **Data-Flow Defects** - reasonable operational sequences that data should flow through



-
- ▶ **Data Defects** - incorrect implementation of data structures
 - ▶ **Module Interface Defects**
 - ▶ **Code Documentation Defects**
 - ▶ code documentation does not reflect what the program actually does, or is incomplete or ambiguous
 - ▶ **External Hardware, Software Interfaces Defects**
 - ▶ problems related to system calls, links to databases, input/output sequences, memory usage, resource usage, interrupts and exception handling, data exchanges with hardware, protocols, formats, interfaces with build files, and timing sequences (race conditions may result).



Testing defects



- ▶ **Test Harness Defects**

- ▶ Auxillary code

- ▶ **Test Case Design and Test Procedure Defects**

- ▶ incorrect, incomplete, missing, inappropriate test cases, and test procedures.



Thank You