TESTING STRATEGIES

LECTURE # 29

LECTURE BY ENGR. SIDRA

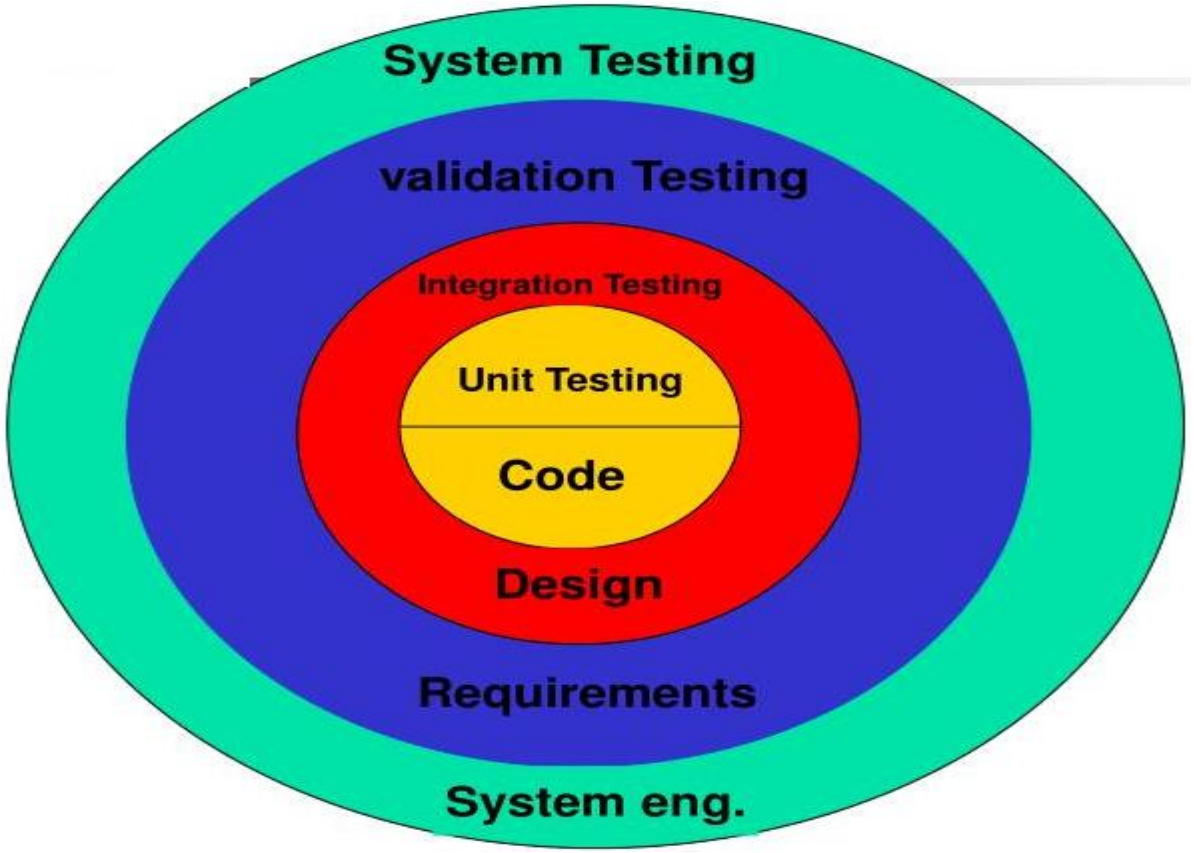
STRATEGIC APPROACH (CHARACTERISTICS)

- To perform effective testing, you should conduct effective technical reviews. By doing this, many errors will be eliminated before testing commences.
- Testing begins at the component level and works "outward" toward the integration of the entire computer-based system.
- Different testing techniques are appropriate for different software engineering approaches and at different points in time.
- Testing is conducted by the developer of the software and (for large projects) an independent test group.
- Testing and debugging are different activities, but debugging must be accommodated in any testing strategy.

TESTING STRATEGY

- There are many strategies that can be used to test software.
- At one extreme, you can wait until the system is fully constructed and then conduct tests on the overall system in hopes of finding errors.
  - This approach, although appealing, simply does not work.
  - It will result in buggy software that disappoints all stakeholders.
- At the other extreme, you could conduct tests on a daily basis, whenever any part of the system is constructed.
  - This approach, although less appealing to many, can be very effective.

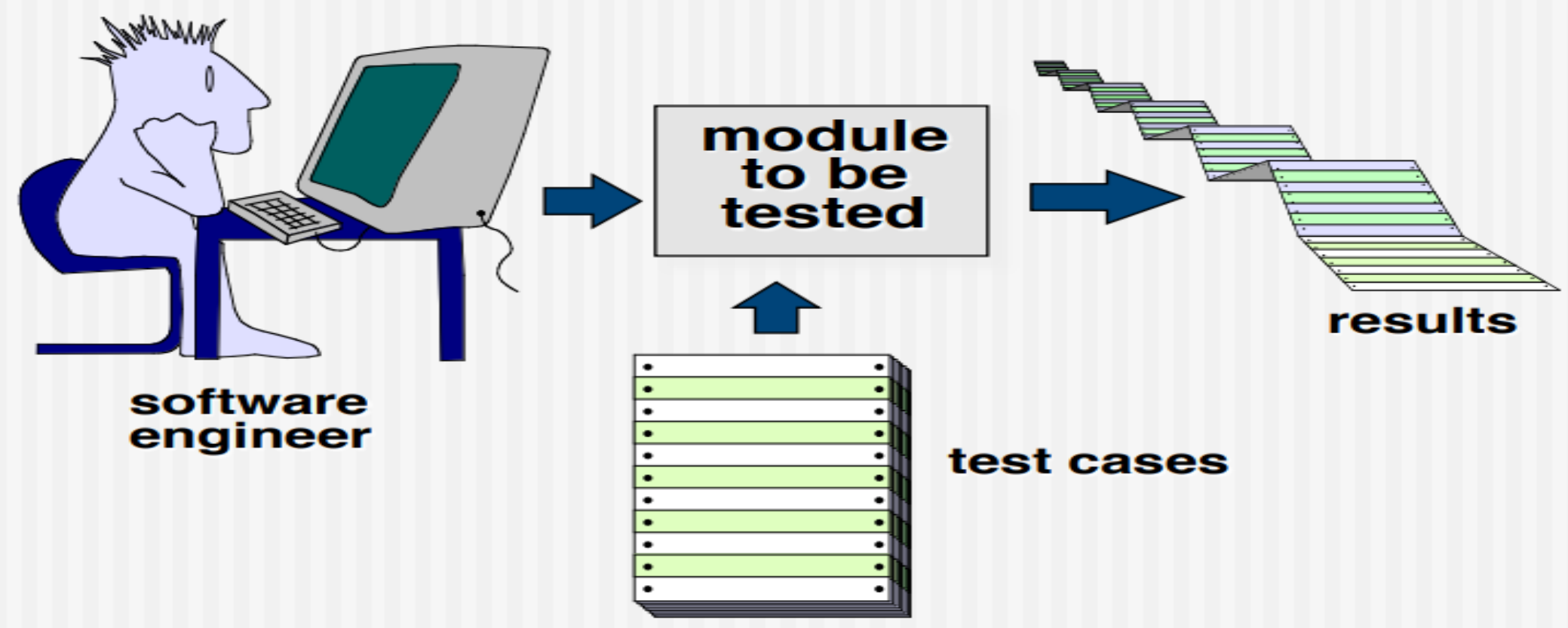
TESTING STRATEGY



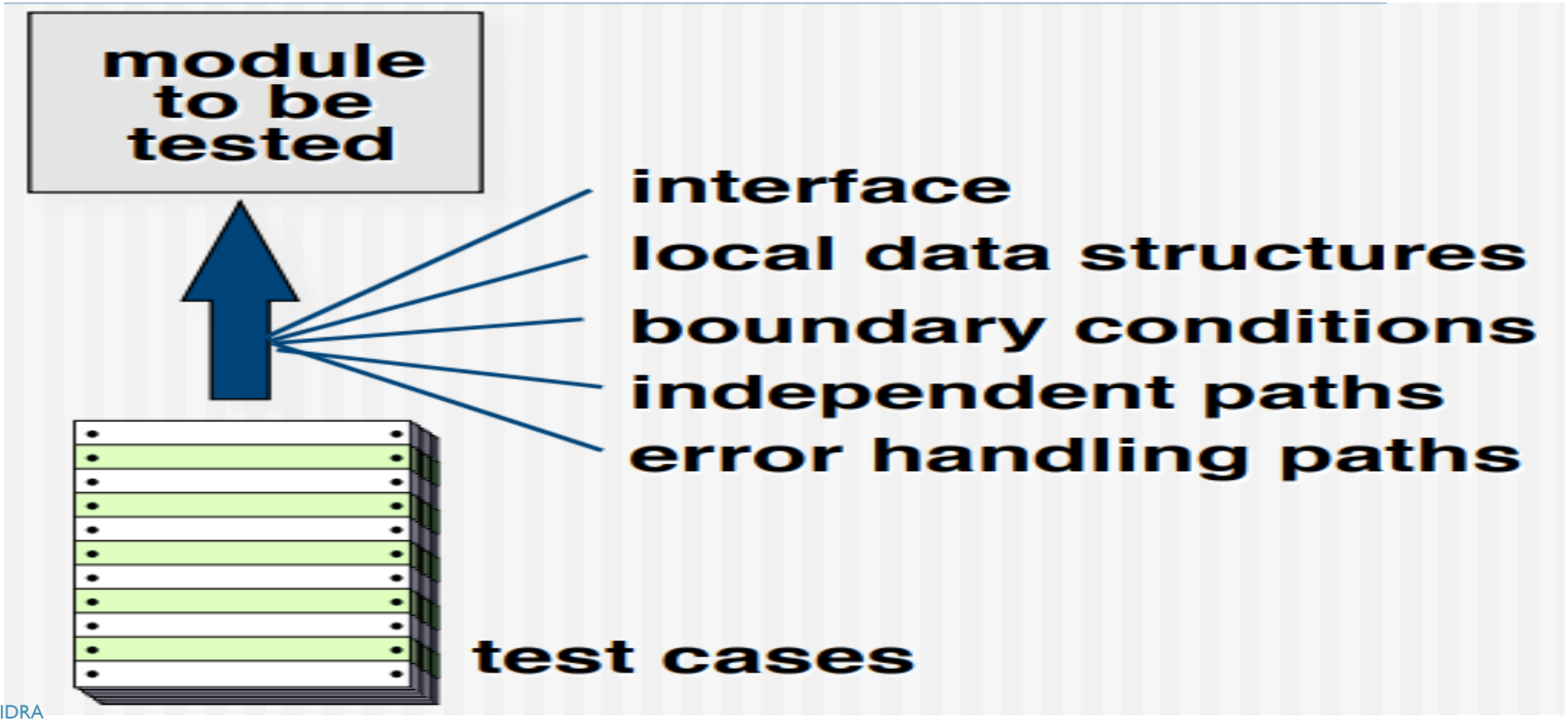
UNIT TESTING

- Unit testing is the process of testing individual components in isolation.
- The purpose is to discover defects.
- Units may be:
  - Individual functions or methods within an object
  - Object classes with several attributes and methods
  - Composite components with defined interfaces used to access their functionality.

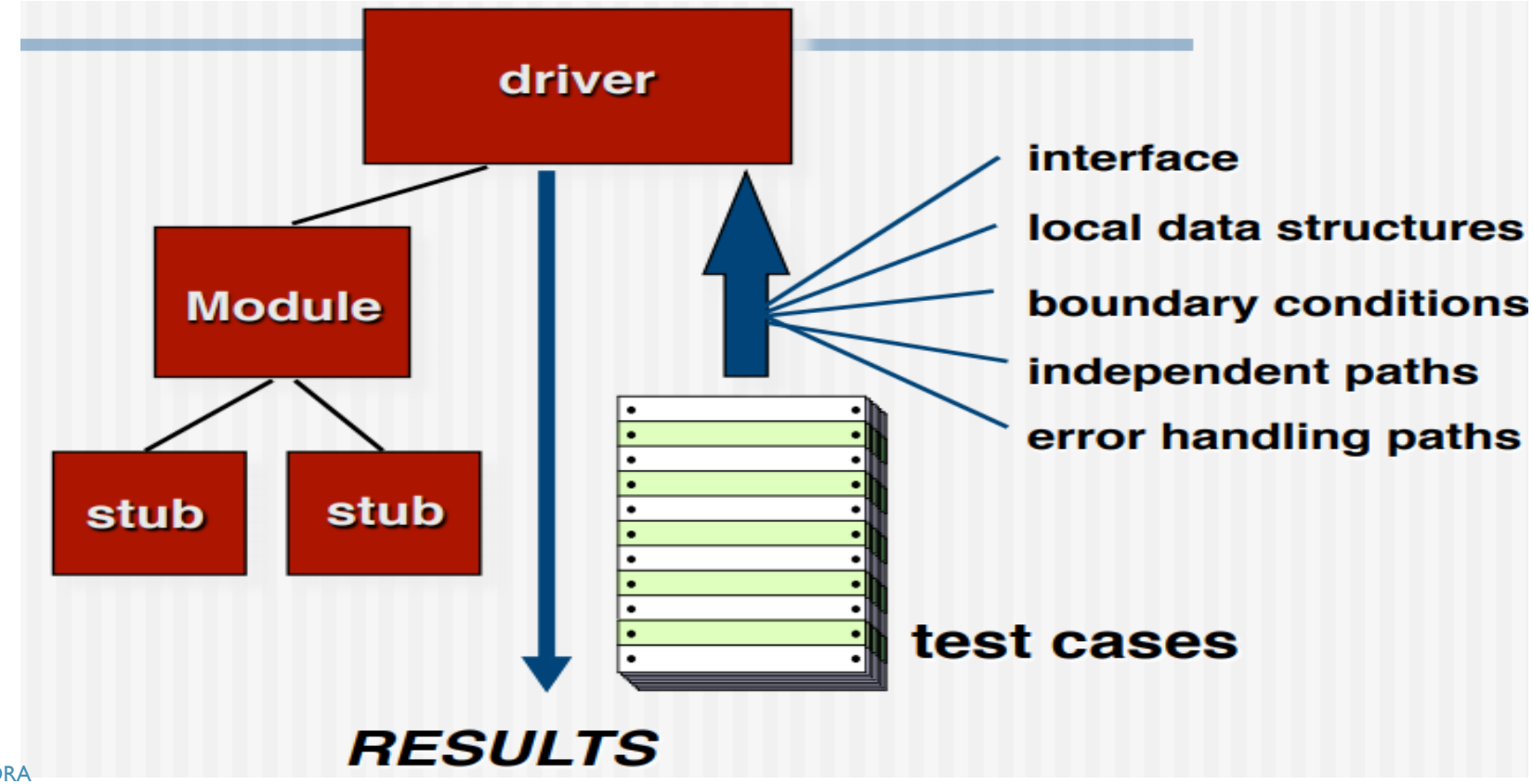
UNIT TESTING



UNIT TESTING



UNIT TEST ENVIRONMENT



INTEGRATION TESTING

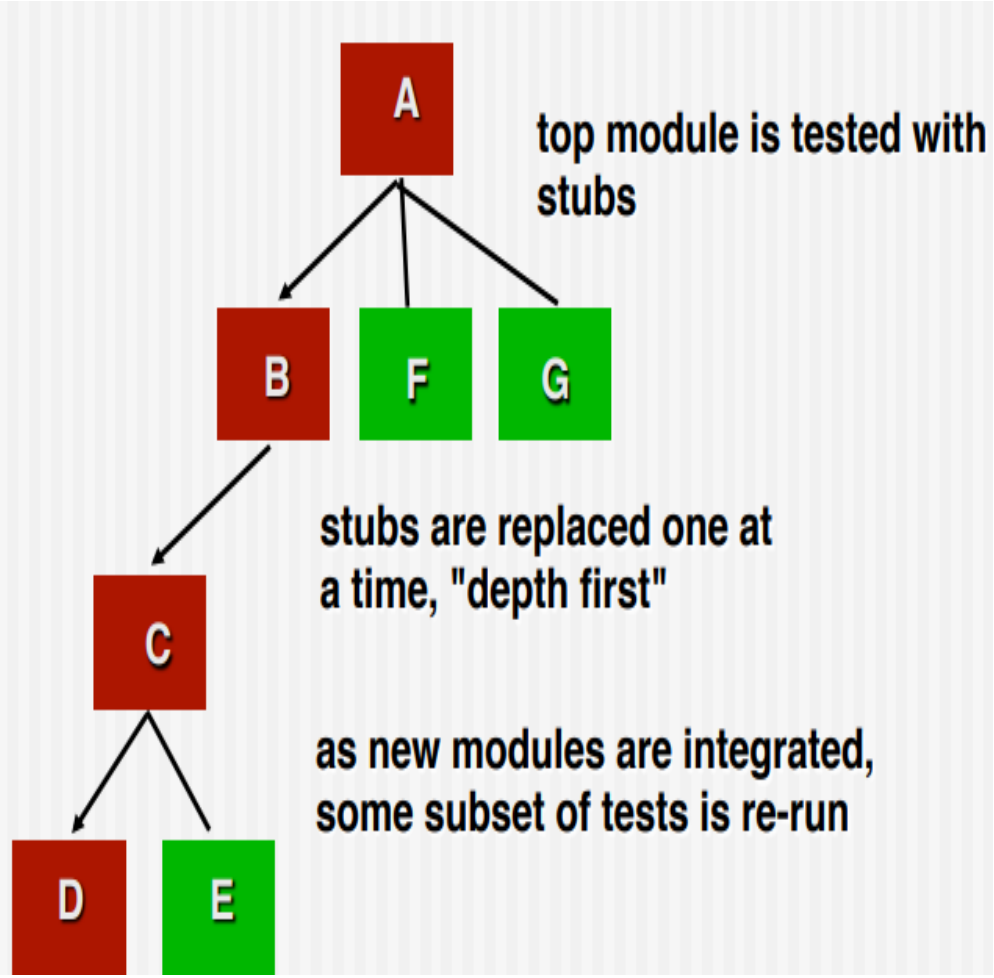
- Options:
  - the "big bang" approach (all at once)
  - an incremental construction strategy

INTEGRATION TESTING STRATEGIES

- Top-down integration
- Bottom-up integration
- Regression testing

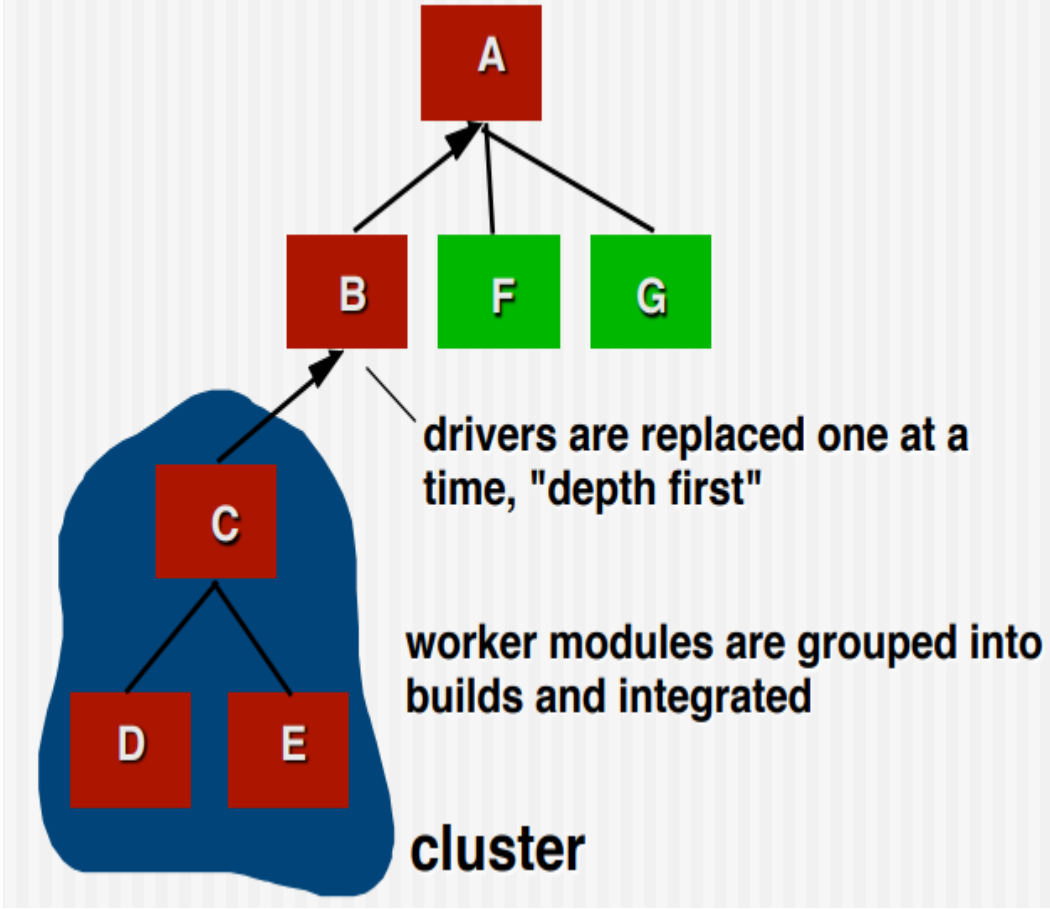
INTEGRATION TESTING STRATEGIES

- Top-down testing
  - Start with high-level system and integrate from the top-down replacing individual components by stubs where appropriate
  - Depth-first vs. breadth-first



INTEGRATION TESTING STRATEGIES

- Bottom-up testing
  - Integrate individual components in levels until the complete system is created
  - In practice, most integration involves a combination of these strategies



INTEGRATION TESTING STRATEGIES

- Regression testing
  - It is the re-execution of some subset of tests that have already been conducted to ensure that changes have not propagated unintended side effects
  - Whenever software is corrected, some aspect of the software configuration (the program, its documentation, or the data that support it) is changed.
  - Regression testing helps to ensure that changes (due to testing or for other reasons) do not introduce unintended behavior or additional errors.
  - Regression testing may be conducted manually, by re-executing a subset of all test cases or using automated capture/playback tools.