of f

Integration Testing

Determing the correctness of interface is the focus of Integration Testing.

Why Integration is required

- Data may be lost during interfacing
- Global data may also cause problem
- Sub function may not work properly when combined

Method to integrate & Test

Unit Component are taken one by one and integrated incrementally

⇓

Debugging & fault isolation become easier

Types of Integration

1. Top-Down:
2. Bottom-Up
3. Sandwich Combination of Top-Down + & Bottom-Down

Top-Down Module : Calling module always available
No drivers required

Bottom-Up Drivers are required but stubs are not required

Top-Down Integration

- * Move down in central hirachy
- * Start with main module & integrate the sub-modules

Dept-first
Integrate the unit on a major control path

Breadth-first
combine all the components direct subordinate to a given module

Integration Steps

1. starting with main ~~control~~ central module stubs are substituted for all components subordinate to it

2. Dependency on type of integration stub are replaced one at a time with actual component

3 Test are conducted as each component integrated

4 One completion another stub replaced with actual component

5 Regression testing ensure no new errors.

Problem Stubs lead to extra effort in form of overhead head. soln

Bottom-up Integration

* Begin testing at lowest level
* Drivers required but No stubs needed

Steps.

1. Low level components are combined into cluster / Builds that perform specific S/w function.
2. Driver coordinates the test case I/P - O/P
3 Cluster is tested
4. Driver is removed & the cluster are combined with new module moving up in hierarchy.

Validation Testing

* focuses on used - visible Action & user recognizable output from the system

* Validation succeeds when software function in a manner that can be expected by the customer.

* Validation criterion : is a

✓ section in [SRS] document which describes the desired functionality the forms the basis of validation testing

How is validation achieved ?

Though a series of test that demonstrate the existence of functionality read by user in SW.

✓ What does Validation Testing Enquire!

* functionality is achieved.
* correct behaviour of is achiever
* Performance constraints met.
* Document are correct
* Various non-functional require (usability etc
* A deficiency list may be ~~test~~ created in case something is missing / incorrect

Alpha Beta (406)

System Testing

- It incoporates software with other system elements
- System Testing ensure that all element system have integrated well with the software & the entire system functions properly.

Types of System Testing

Recovery Testing

It must ensure that system must recover from faults & resume processing with little or no downtime

Aim: To develop a fault tolerant system Any processing fault should not bring down the entire system

- S/w must recover & resume its normal process within specified timeperiod

How it is Done?

- We intentionally force the system to fail different ways & then we verify that the recovery of system is done properly.
- If the recovery is automated reinitialization, checkpointing mechanism, data recovery & restart are evaluated for correctness
- If recovery require human intervention the mean-time-to-repair (MTTR) is evaluated to determine whether it is within acceptance limits

Security Testing

verify that a protection mechanism built into system will actually protect it from attacks.

Responsibility of tester

↙ Act as an intruder

⇘ ensures that cost of attack is higher than information obtained.

Stress Testing

It execute the system in manner that demands resources in abnormal quantity

E.g. (10) per min

↳ excessive memory reqmts

→ database request

→ simultaneous log-in

Performance testing

* Aimed at testing the runtime performance of software

* Ensure that performance is monitore cont. beginning from testing itself