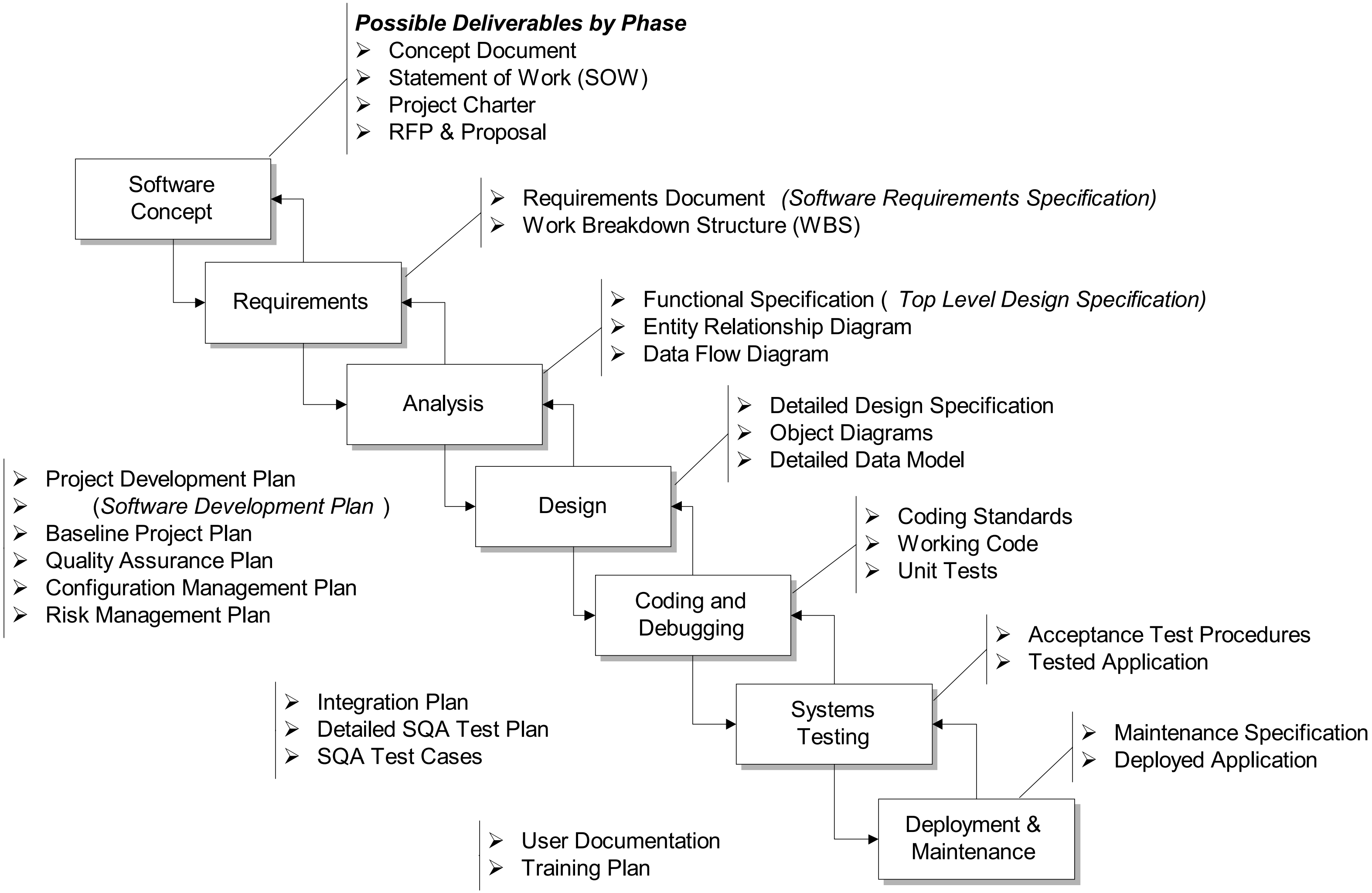
Integration and Testing

Today

- Software Quality Assurance
- Integration
- Test planning
- Types of testing
- Test metrics
- Test tools

Deliverables by Phase



Possible Deliverables by Phase

- Concept Document
- Statement of Work (SOW)
- Project Charter
- RFP & Proposal

Software Concept

- Requirements Document *(Software Requirements Specification)*
- Work Breakdown Structure (WBS)

Requirements

- Functional Specification (*Top Level Design Specification)*
- Entity Relationship Diagram
- Data Flow Diagram

Analysis

- Detailed Design Specification
- Object Diagrams
- Detailed Data Model

Design

- Project Development Plan
- (*Software Development Plan*)
- Baseline Project Plan
- Quality Assurance Plan
- Configuration Management Plan
- Risk Management Plan

- Coding Standards
- Working Code
- Unit Tests

Coding and Debugging

- Acceptance Test Procedures
- Tested Application

- Integration Plan
- Detailed SQA Test Plan
- SQA Test Cases

Systems Testing

- Maintenance Specification
- Deployed Application

Deployment & Maintenance

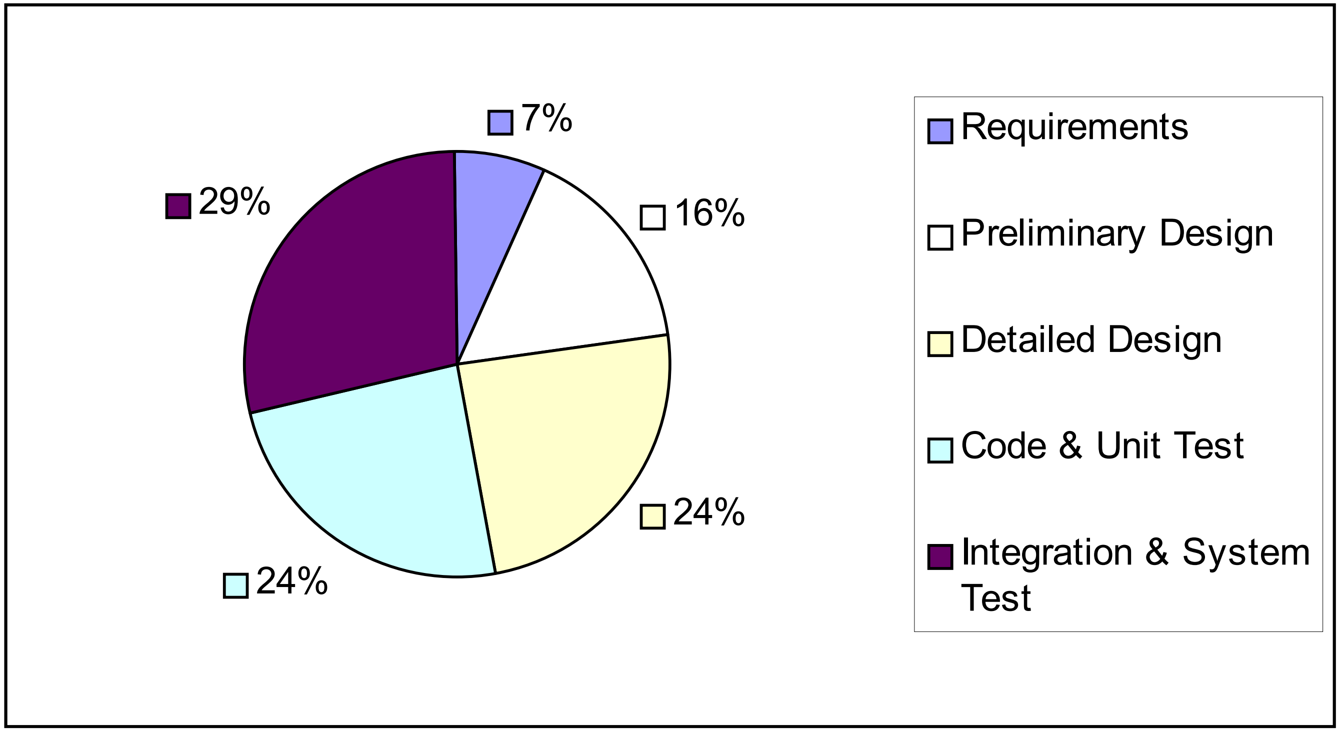
- User Documentation
- Training Plan

If 99.9% Were Good Enough

- 9,703 cheques would be deducted from the wrong bank accounts each hour
- 27,800 pieces of mail would be lost per hour
- 3,000,000 incorrect drug prescriptions per year
- 8,605 commercial aircraft takeoffs would annually result in crashes

Futrell, Shafer, Shafer, “Quality Software Project Management”, 2002

Development Costs



Integration & Testing

- Development/Integration/Testing
  - Common place for schedule & activity overlap
- Sometimes Integration/Testing thought of as one phase
- Progressively aggregates functionality
- QA team works in parallel with development team

Integration Approaches

Top Down

- Core or overarching system(s) implemented first
- Combined into minimal “shell” system
- “Stubs” are used to fill-out incomplete sections
    - Eventually replaced by actual modules

Bottom Up

- Starts with individual modules and builds up
- Individual units (tested) combined into sub-systems
- Sub-systems are combined into the whole

Integration

Who does integration testing?

- Can be either development or QA team

Staffing and budget are at peak

Issues

- Pressure
- Delivery date nears
- Unexpected failures (bugs)
- Motivation issues
- User acceptance conflicts

Validation and Verification

- V & V
- Validation
  - Are we building the right product?
- Verification
  - Are we building the product right?
  - Testing
  - Inspection
  - Static analysis

Quality Assurance

- QA or SQA (Software Quality Assurance)
- Good QA comes from good process
- When does SQA begin?
  - During requirements
- A CMM Level 2 function

QA Roles

- QA Manager
  - Hires QA team; creates test plans; selects tools; manages team
- Test Developer/Test Engineer
  - Performs functional tests; develops automated scripts
- System Administrator
  - Supports QA functions; not official QA team member
- Copy Editor/Documentation Writer
  - Supports QA; also not part of official team

Test Plans (SQAP)

- Software Quality Assurance Plan
  - Should be complete near end of requirements
- See example
  - Even use the IEEE 730 standard

SQAP

Standard Sections

- Purpose
- Reference documents
- Management
- Documentation
- Standards, practices, conventions, metrics
  - Quality measures
  - Testing practices

SQAP ..2

- Reviews and Audits
  - Process and specific reviews
    - Software Requirements Review (SRR)
    - Test Plan Review
    - Code reviews
    - Post-mortem review
- Risk Management
- Problem Reporting and Corrective Action
- Tools, Techniques, Methodologies
- Records Collection and Retention

Software Quality

- Traceability
  - Ability to track relationship between work products
  - How well do requirements/design/test cases match?
- Formal Reviews
  - Conducted at the end of each lifecycle phase
  - Software Requirements Review (SRR)
  - Design Review
  - Code Review
  - And so on

Testing

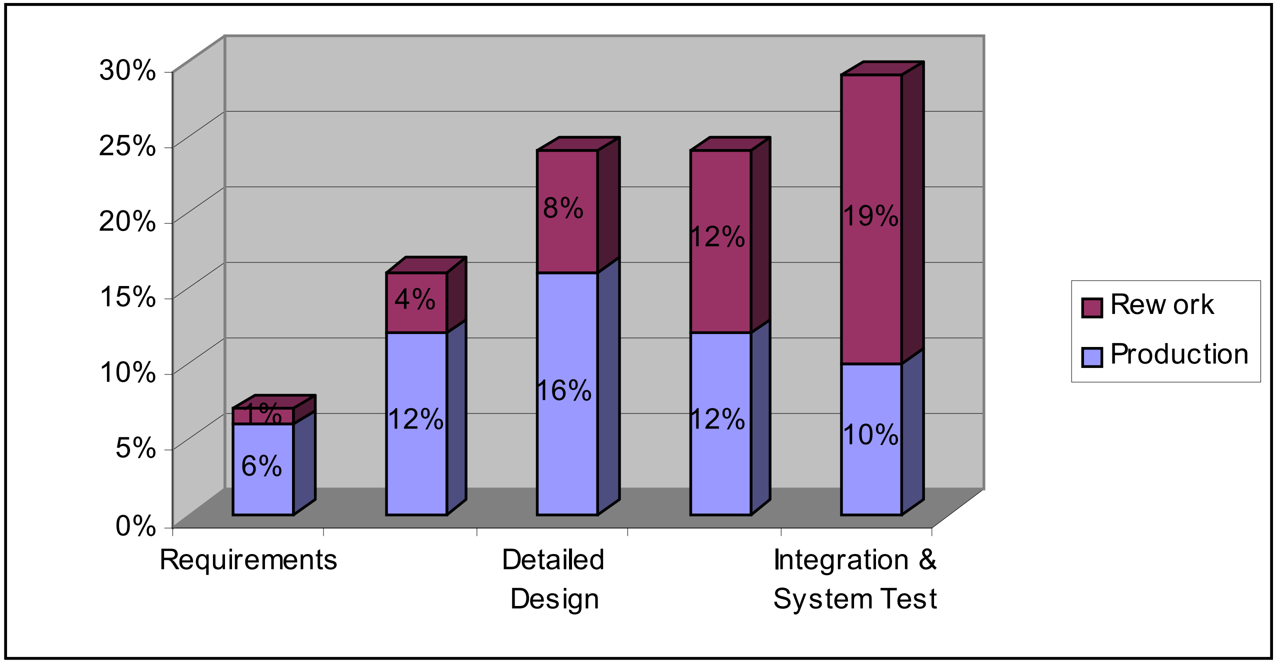
- Exercising computer program with predetermined inputs
- Comparing the actual results against the expected results
- Testing is a form of sampling
- Cannot absolutely prove absence of defects
- Testing is not debugging

Test Cases

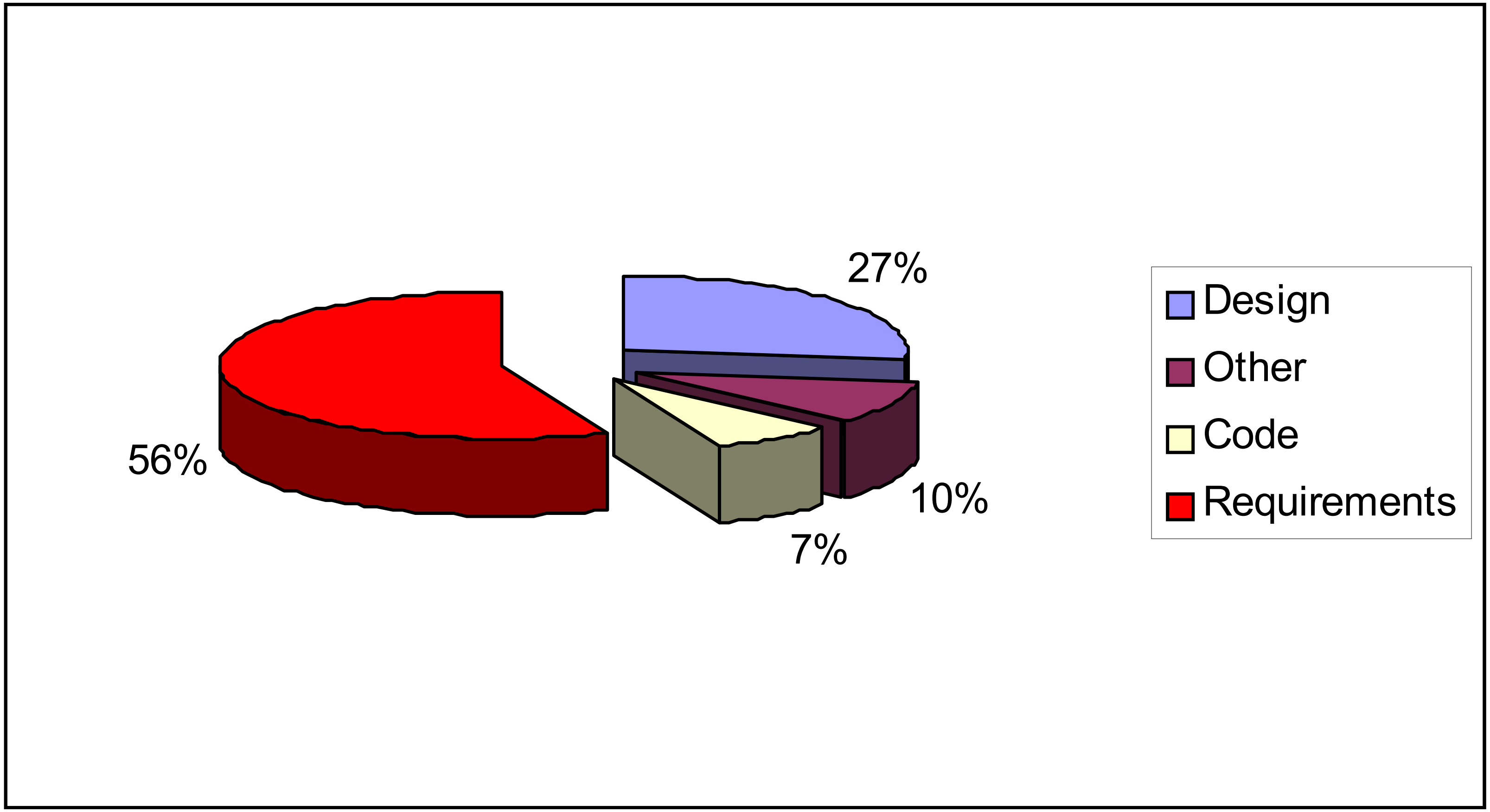
- Key elements of a test plan
- May include scripts, data, checklists
- May map to a Requirements Coverage Matrix
    - A traceability tool

Rework

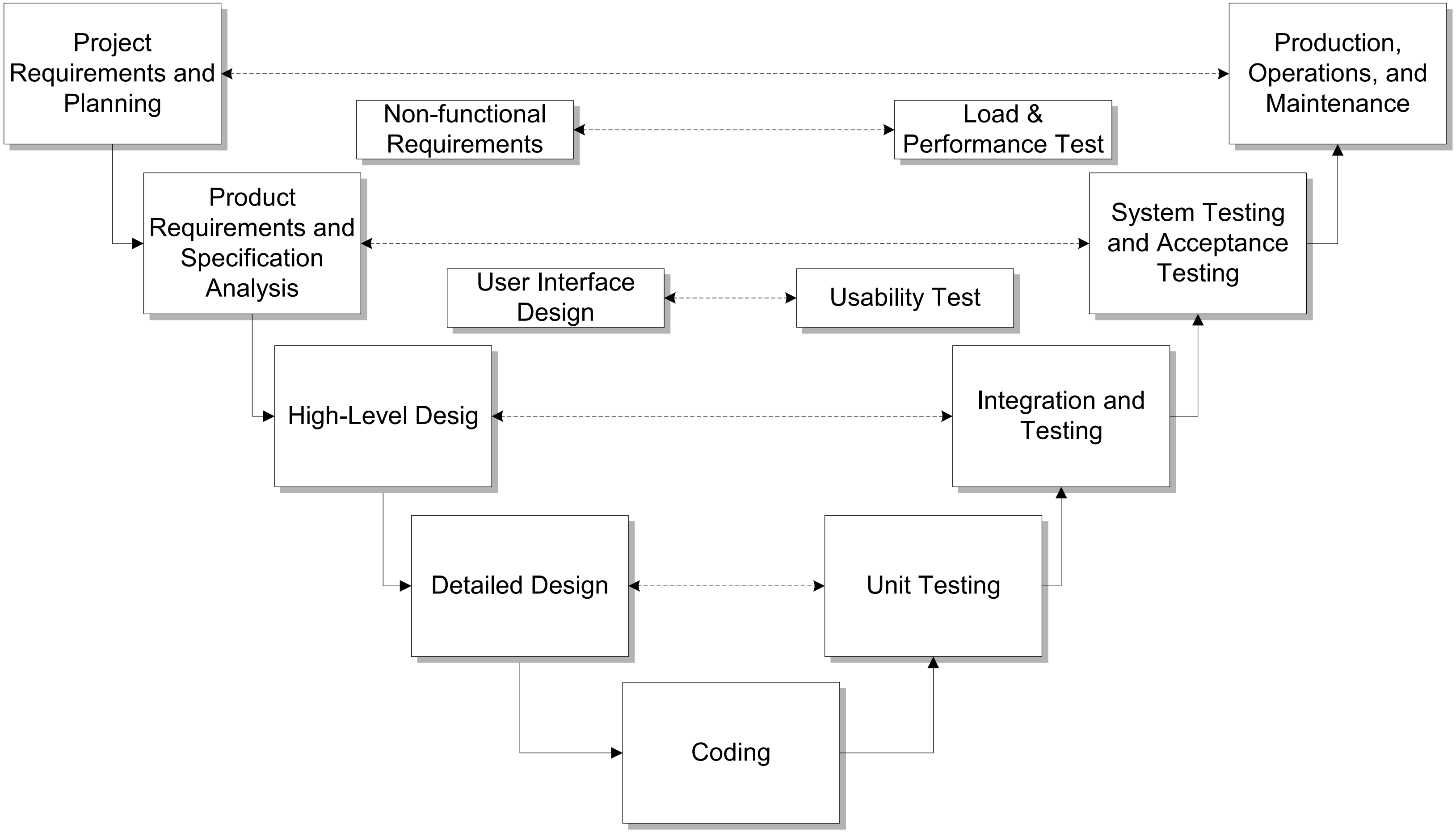
- Software equivalent of “scrap” in manufacturing



Sources of Defects



V Process Model


Project Requirements and Planning
Production, Operations, and Maintenance
Non-functional Requirements
Load & Performance Test
Product Requirements and Specification Analysis
System Testing and Acceptance Testing
User Interface Design
Usability Test
High-Level Desig
Integration and Testing
Detailed Design
Unit Testing
Coding

Project Testing Flow

- Unit Testing
- Integration Testing
- System Testing
- User Acceptance Testing

Black-Box Testing

- Functional Testing
- Program is a “black-box”
  - Not concerned with how it works but what it does
  - Focus on inputs & outputs
- Test cases are based on SRS (specs)

White-Box Testing

- Accounts for the structure of the program
- Coverage
  - Statements executed
  - Paths followed through the code

Unit Testing

- a.k.a. Module Testing
- Type of white-box testing
  - Sometimes treated black-box
- Who does Unit Testing?
  - Developers
  - Unit tests are written in code
- When do Unit Testing?
  - Ongoing during development
  - As individual modules are completed

Unit Testing

- Individual tests can be grouped
  - “Test Suites”
- JUnit
    - Part of the XP methodology
    - “Test-first programming”

Integration Testing

- Testing interfaces between components
- First step after Unit Testing
- Components may work alone but fail when put together
- Defect may exist in one module but manifest in another
- Black-box tests

System Testing

- Testing the complete system
- A type of black-box testing

User Acceptance Testing

- Last milestone in testing phase
- Ultimate customer test and sign-off
- Sometimes synonymous with beta tests
- Customer is satisfied software meets their requirements
- Based on “Acceptance Criteria”
  - Conditions the software must meet for customer to accept the system
  - Ideally defined before contract is signed
  - Use quantifiable, measurable conditions

Regression Testing

- Re-running of tests after fixes or changes are made to software or the environment
- Example
  - QA finds defect
  - Developer fixes
  - QA runs regression test to verify
- Automated tools very helpful for this

Compatibility Testing

- Testing against other platforms
  - Testing against multiple browsers
  - Does it work under Netscape/IE, Windows/Mac?
  - Testing against multiple operating systems

External Testing Milestones

- Alpha 1st, Beta 2nd
- Testing by users outside the organization
- Alpha release
  - Given to very limited user set
  - Product is not feature-complete
- Beta release
  - Customer testing and evaluation
  - Preferably after software stabilizes

External Testing Milestones

- Value of Beta Testing
  - Testing in the real world
  - Getting a software assessment
  - Marketing
  - Augmenting your staff
- Beta testers must be “recruited”
  - Existing base, marketing, tech support, site
- All testing must be scheduled by PM

External Testing Milestones

- Release Candidate (RC)
  - Send to manufacturing, if testing successful
- Release to Manufacturing (RTM)
  - Production release formally sent
- Aim for a ***stabilization period*** before each of these milestones
  - Team focus on quality, integration, stability

Test Scripts

Two meanings:

1. Set of step-by-step instructions intended to lead test personnel through tests
   - List of all actions and expected responses

2. Automated test script (program)

Static Testing

Reviews

- Most artifacts can be reviewed
- Proposal
- Schedule
- Requirements
- Data model
- Code
- Test plans

Static Testing ..2

Peer Reviews

- Methodical examination of software work products by peers to identify defects and necessary changes
- Goal: remove defects early and efficiently
- Planned by PM, performed in meetings, documented
- CMM Level 3 activity

Automated Testing

Human testers are inefficient

Pros

- Lowers overall cost of testing
- Tools can run unattended
- Tools run through ‘suites’ faster than people
- Great for regression and compatibility tests
- Tests create a body of knowledge
- Can reduce QA staff size

Automated Testing ..2

Cons

- Not everything can be automated
- Learning curve or expertise in tools
- Cost of high-end tools $5-80K (low-end are still cheap)

Test Tools

- Capture & Playback
- Coverage Analysis
- Performance Testing
- Test Case Management

Load & Stress Testing

Push system beyond capacity limits

Often done via automated scripts

- By the QA team
- Near end of functional tests

Can show

- Hidden functional issues
- Maximum system capacity
- Unacceptable data or service loss
- Determine if “Performance Requirements” met

Load & Stress Testing ..2

Metrics

- Minimal acceptable response time
- Minimal acceptable number of concurrent users
- Minimal acceptable downtime

High-end Vendors

- Segue
- Mercury
- Empirix

Performance Metrics

Bad	**Good**
Must support 500 users	Must support 500 simultaneous users
10 second response time	[Average|Maximum|90th percentile] response time must be X seconds
Must handle 1M hits per day	Must handle peak load of 28 page requests per second

Source: Athens Consulting Group

Other Testing

Installation Testing

- Very important if not a Web-based system
- Can lead to high support costs and customer dissatisfaction

Usability Testing

- Verification of user satisfaction
    - Navigability
    - User-friendliness
    - Ability to accomplish primary tasks

Other Issues

Pareto Principle

- The 80-20 rule (80% of defects from 20% of code)
- Identify the problem modules

Phase Containment

- Testing at the end of each phase
- Prevent problems moving phase-to-phase

Burn-in

- Allowing system to run long period of time
- Variation of stress testing

Other Issues ..2

Code Freeze

- When developers stop writing new code and only do bug fixes
- Occurs at varying points in integration/testing

Tester-to-Coder Ratio

- Often 1:3 or 1:4
- QA staff and Manager may be needed

Stopping Testing

- When do you stop?
- Rarely are all defects “closed” by release
- Shoot for all Critical/High/Medium defects
- Often, occurs when time runs out
- Final Sign-off (User Acceptance Test)
  - Customers
  - Engineering
  - Product management

Test Metrics

Load

- Maximum acceptable response time
- Minimum number of simultaneous users

Disaster

- Maximum allowable downtime

Compatibility

- Minimum and maximum number of browsers and operating systems supported

Test Metrics ..2

Usability

- Minimum approval rating from focus groups

Functional

- Requirements coverage
- 100% pass rate for automated test suites

Defect Metrics

These are very important to the PM

Number of outstanding defects

- Ranked by severity
    - Critical, High, Medium, Low
    - Showstoppers

Opened vs. closed

Defect Tracking

Get tools to do this for you

- Bugzilla, TestTrack Pro, Rational ClearCase

Make sure all necessary team members have access

Have regular ‘defect review meetings’

- Can be weekly early in test, daily in crunch

Who can enter defects into the tracking system?

- Lots of people: QA staff, developers, analysts, managers, (sometimes) users, PM

Defect Tracking ..2

Fields

- State: open, closed, pending
- Date created, updated, closed
- Description of problem
- Release/version number
- Person submitting
- Priority: low, medium, high, critical
- Comments: by QA, developer, other

Defect Metrics

Open Rates

- How many new bugs over a period of time?

Close Rates

- How many closed over that same period?
- Ex: 10 bugs/day

Change Rate

- Number of times the same issue updated

Fix Failed Counts

- Fixes that didn’t really fix (still open)

Web Site Testing

Unique factors

- – Distributed (N-tiers, can be many)
- – Very high availability needs
- – Uses public network (Internet)
- – Large number of platforms (browsers and OS)

Web Site Testing ..2

Causes of most site failures

- Internal network performance
- External network performance
- Hardware performance
- Unforeseeable traffic spikes
- Web application performance

Web Site Testing

- Commercial Tools: Load Test & Site Management
  - Mercury Interactive
    - SiteScope, SiteSeer
  - Segue
- Commercial Subscription Services
  - Keynote Systems
- <u>Monitoring Tools</u>