



# The Project Academy Series

## Test Management

May 27, 2015 and May 29, 2015

8:30 AM - 12:00 PM



# Welcome and Introductions

**Janet Fong**

**IT Project Advisor - Testing**

**Consulting and Planning Division**

**Andy Wertin**

**IT Project Advisor - Governance, Risk and Issue**

**Management, and Organizational Change Management**

**Consulting and Planning Division**

# Objectives

- Understand the purpose and importance of testing.
- Obtain high level knowledge of the Test Process Activities.
- Learn common pitfalls and tips for addressing test challenges.

# Agenda

- Testing Concepts
  - What is Testing?
  - Why is Testing Important?
  - Cost of Defects
- Test Process Activities, Common Test Pitfalls, and Tips
- **Break & Exercise**
- Summary
- Question and Answers

# What is Testing?

- Testing = Planning, preparation, & evaluation of software products & work products to determine that they satisfy specified requirements, to demonstrate they do the job it was designed to do, & to detect defects.

- Requirement & Testing partners.

- Testing relies on testable requirements.



- Test Management = Management of project test activities and efforts across SDLC.

# Why is Testing Important?

- Information Technology is part of our daily lives.



*“I always remind people that in our personal lives, technology’s becoming a bigger and bigger part of the way we do things. Consumers are bringing those expectations to government. They want government to be accessible around the clock, they want it to be easy to navigate.”*

*Statescoop interview with Carlos Ramos (4/23/15).*

# Why is Testing Important? (Continue)

- Quality products deliver customer confidence.
  - Quality is the degree to which a product or process meets established requirements.
    - Testing = Identify defects.
    - Quality Control = Verify software to requirements.
    - Quality Assurance = Ensure processes & standard.
  - Quality is planned and built into the system.
- Computer failures from system bugs impact individuals financially or business reputation.

**Free Airline Tickets**



**Savings & Checking  
Accounts - \$146 Million**

# What Do These Amounts Have In Common?



**\$96 Million**

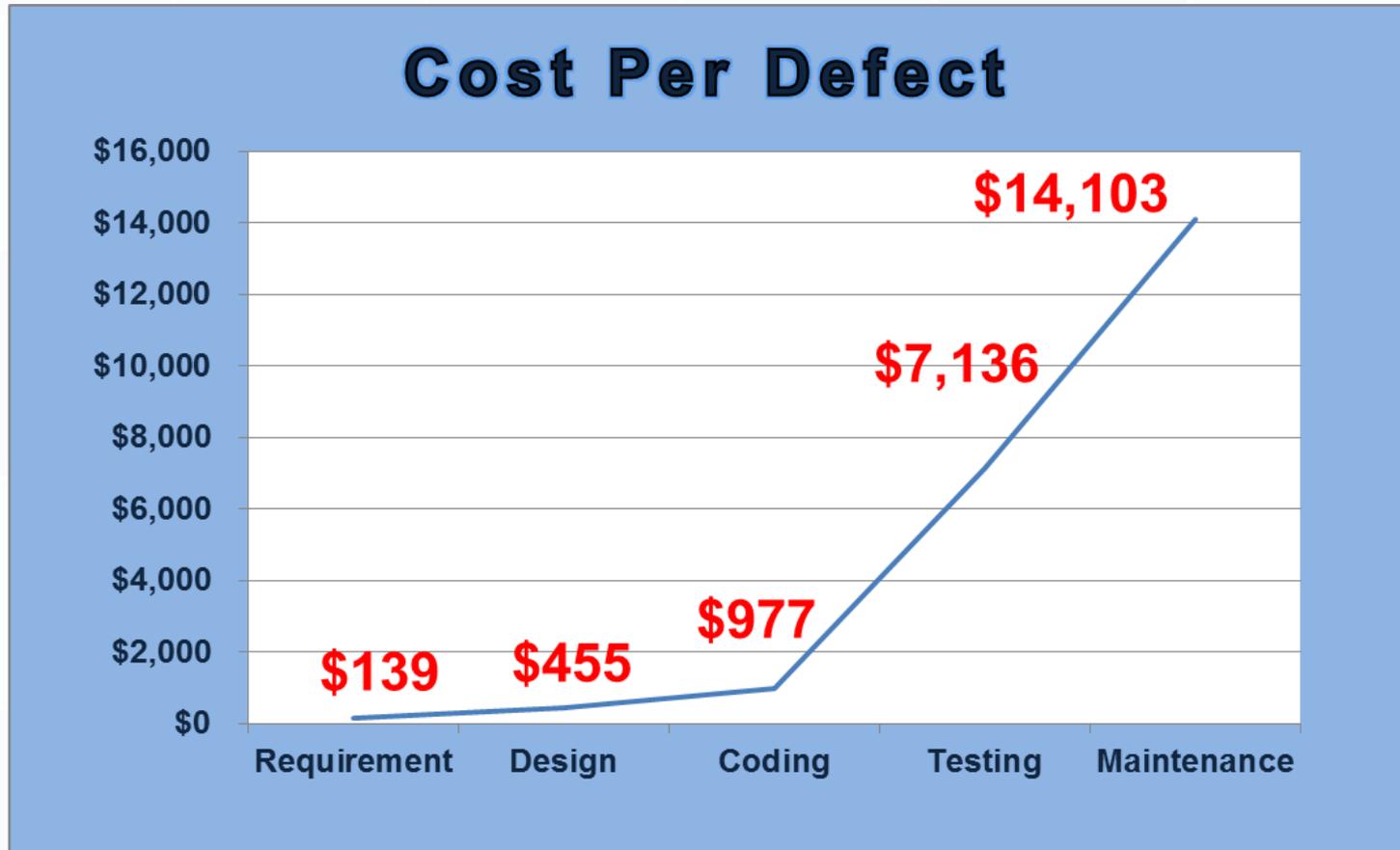
**\$208 Million**



**\$262 Million**



# Cost Per Defect



Source: Caper Jones, [Software Assessments, Benchmarks, and Best Practices](#), Addison-Wesley, 2000.

# Cost of Defects – Continue

## ■ Defect Introduction <sup>1</sup>

■ 20% - Requirements

■ 25% - Design

■ 55% - Code

## ■ Multiplicative cost increase for fixing defects later in the lifecycle. <sup>2</sup>

System	Cost Increase Requirements to Production
Simple	1:5
Complex	1:100

*1 Source: Caper Jones (2008)*

*2 Source: Boehm*



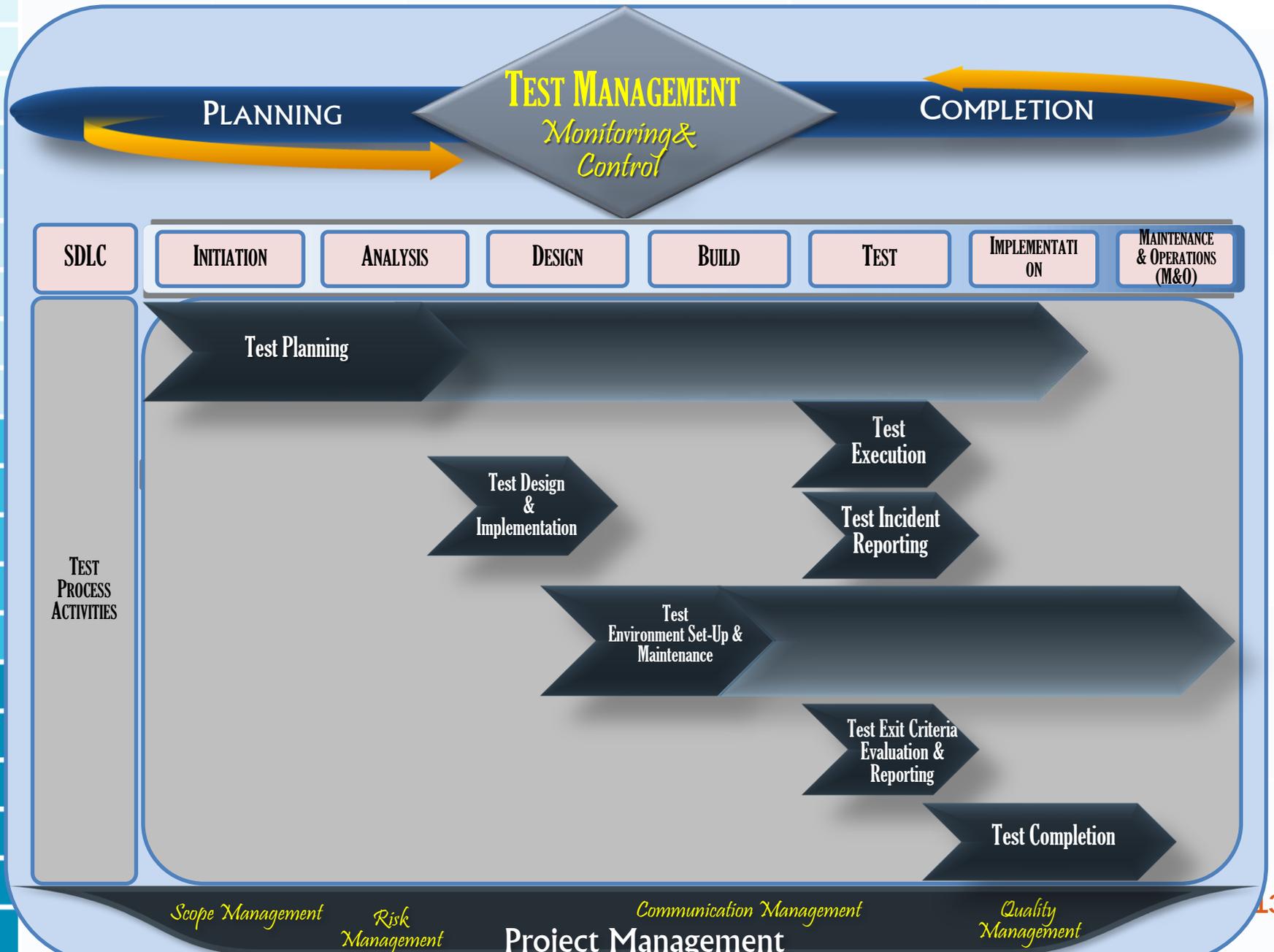
# Test Process Activities



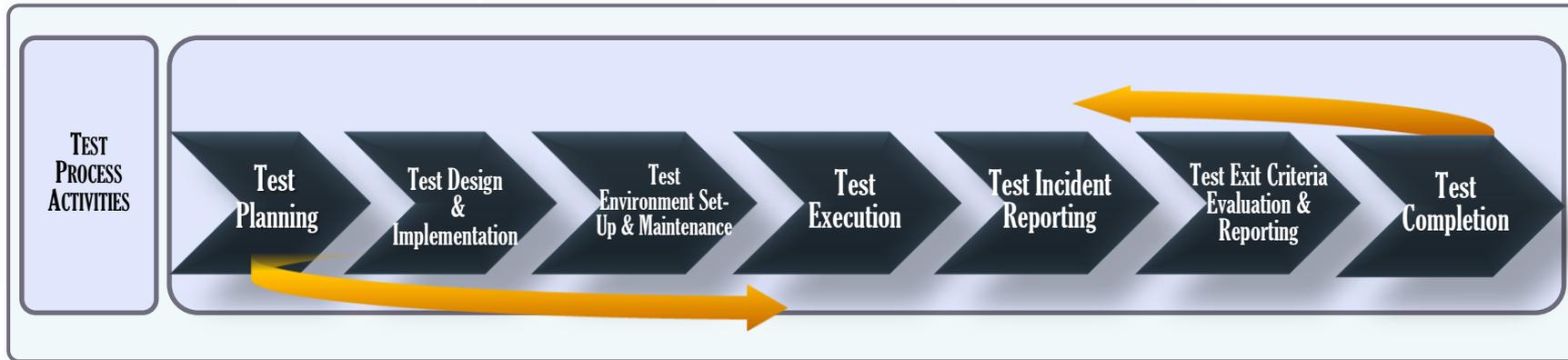
*Begin with an end in  
mind.*

*Stephen Covey*

# When Should Testing Occur In The SDLC?



# Test Process Activities Illustration



- The test process must be consistently followed.
- The test processes are executed iteratively.
- The test process applies across test levels (e.g., Unit, System, UAT).
- The State manages the test activities & deliverables across the SDLC.

# Test Planning Activities

- Test Planning addresses how to achieve the test mission and objectives for a project.
- The test plan is produced in this activity.
  - Test Plan development begins during requirements identification.
  - Examples test plan areas: test strategy, scope, entrance/exit criteria, resources, risk, hardware, software, & environments.
- Key stakeholders need to be involved.
- Test Planning starts at the beginning of the test process and continues until project completion.

# Test Planning Activities Pitfalls

- **The test plan is developed, but not followed across the SDLC.**
  - **Result: Project test challenges, inefficiencies, or delays. During production, testing will likely be unstructured causing difficulty supporting business needs timely.**
- **Level test plans are not developed (e.g., System, UAT).**
  - **Result: Unplanned test complexity, resources issues, longer time for testing, and increased project cost.**
- **Test Plan is not approved and baselined.**
  - **Result: Business gets a poorly tested product because testing is a “moving target”.**

# Test Planning Activities Tips

- Requirement tasks in the contract for developing the test plans (e.g., master, system).
- Initiate contact early within department to secure resources.
- Key stakeholders agree on the entrance/exit criteria & the measurements for whether the criteria have been met.
- State manages & approves test plan activities, tasks, and deliverables throughout SDLC (e.g., change control).
- Test plan(s) is(are) baselined because it sets the foundation for testing (**Important**).



# Break & Exercise

# Test Design and Implementation Activities

- Test Design and Implementation is the heart of testing.
- The activities produced in this process include:

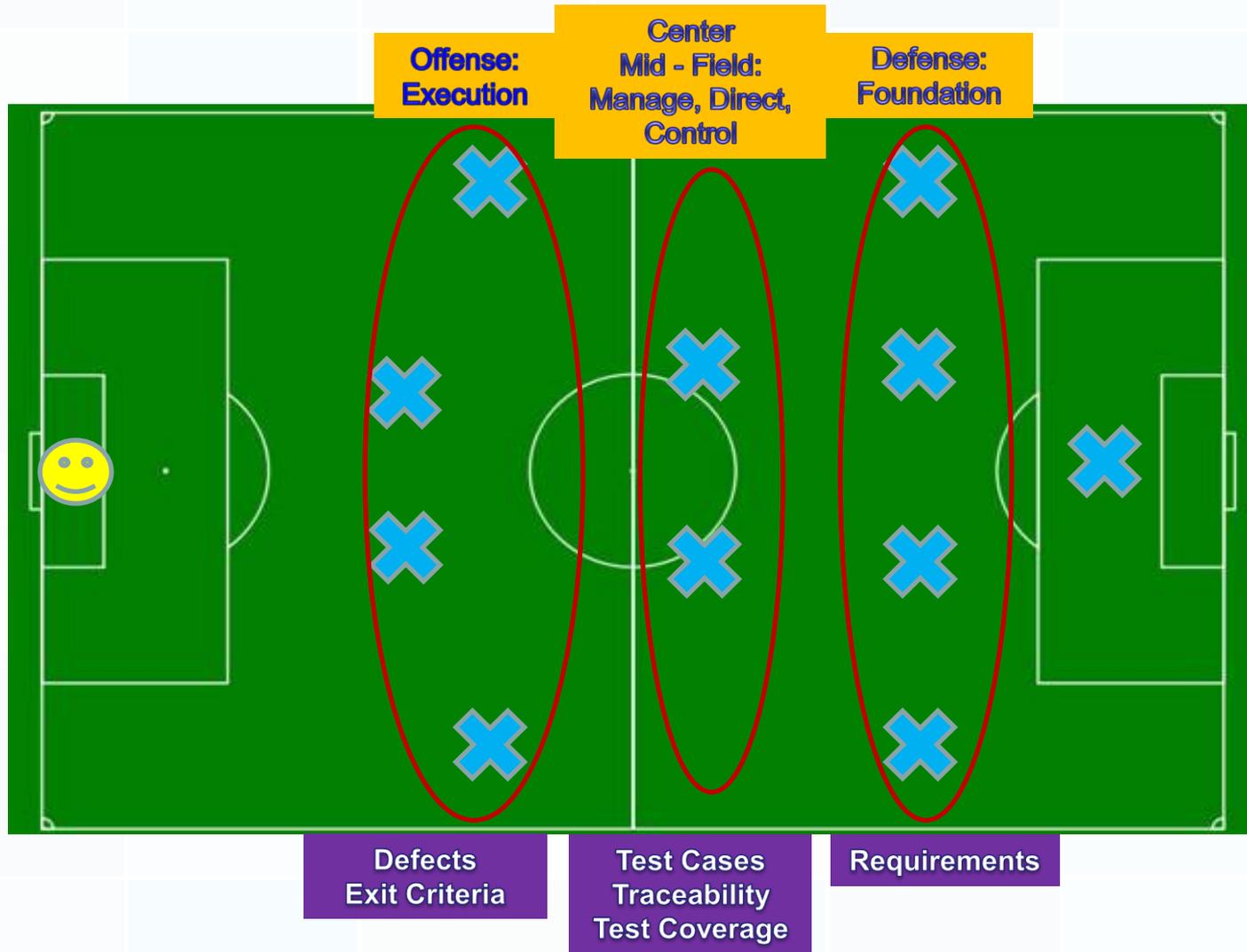
- **Test Traceability (Important)**

- Maps relationship between requirements & test cases that verify the requirements. Important during UAT.



- Test Cases – what will be executed and covered.
- Test Procedures (scripts) – how test cases are executed.
- Test Coverage - the degree of features tested is covered.
- Test Data and Environments Requirements.

# Example - Soccer



# Test Design and Implementation Pitfalls

- **Projects don't spend enough time designing how they plan to test the system.**
  - **Result: The business could receive a system that is not adequately tested (e.g., rich in defects and issues).**
- **Lack of Traceability.**
  - **Result: Challenges occur managing the test progress. The business may receive a production system where there is missing functionality or functionality not tested.**
- **Test cases and procedures are not reviewed and approved.**
  - **Result: This could result in disagreements and lack of business buy-in on what is tested and the priority.**

# Test Design and Implementation Tips

- Build schedule activities for collaborative discussions on test design strategies for validating the software.
  - Examples: Mind Map, Brainstorming, Post-It notes.
- Traceability is very important for efficient test progress management.
  - Test Traceability Matrix on the CPD website.
- Critical Test Design & Implementation artifacts are reviewed, approved, & baselined by key stakeholders. Facilitate test execution (e.g., test cases, procedures, traceability) **(Important)**.

# Test Environment Set-Up and Maintenance Activities

- This activity establishes & maintains the required test environments & data for Test Execution.
- The status of the test environment & data readiness determines when the system can be validated against requirements in Test Execution.
- The prepared test environment, data, and readiness reports are produced in this activity.

# Test Environment Set-Up and Maintenance Pitfalls

- Test environment not reflective of target production environment.
  - Result: Test could yield inaccurate results. Interfaces may not be properly tested.
- Converted test data not verified early enough and done in test execution (e.g., System, UAT).
  - Result: Inaccurate test results or invalid data.
- Resources are not ready to support the test efforts.
  - Result: This could result in test schedule delays.

# Test Environment Set-Up and Maintenance Tips

- Environment closely mirrors target environment.
  - Document gaps, assess risk, & mitigation strategy.
  - Perform cursory test (e.g., smoke test) of environment, data, and processes before start of Test Execution.
- Test data is complete before Test Execution begins.
  - Include criteria for evidence that test data is set-up properly and verified for completeness.
- Department staff committed & ready to support testing.
  - Revisit OLA and SLA terms and agreements.
  - Train internal technical and business staff (e.g., test tools, infrastructure, release & configuration management).

# Test Execution Activities

- Test procedures executed in the test environment, expected test results are compared to actual results and recorded.
- Activity begins after the Test Environment Set-Up and Maintenance process completes.
- In this activity, the Test Execution Log is produced, which records the details of the execution of the test procedures.

# Test Execution Pitfalls

- Test procedures are ambiguous and vague.
  - Results: Different interpretations of the test to be execute and different test results depending on the tester.
- Lack of skilled and experienced testers.
  - Results: Test inconsistencies and inaccurate results.

# Test Execution Tips

- Establish entrance criteria in Test Execution for reviewed & baselined test design artifacts to reduce ambiguous and vague test cases.
- Train test team on test concepts & test processes.
  - Train early in the project on test methodology, processes, and test concepts.
  - Ensure test team understands Test Execution activities & expectations (e.g., communicating test issues, defects, results, and progress).



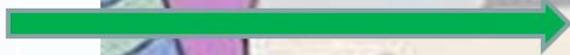
# Exercise Discussion

# Test Incident Reporting Activities

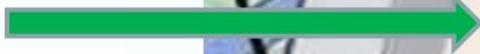
- Activity for reporting to stakeholders issues requiring action identified during Test Execution.
- Test results are analyzed to determine the cause of the incident and appropriate action is taken.
- Activity occurs in parallel with Test Execution activities.
- The Test Incident Report (i.e., defect or bug report) is generated in this process. A Test Incident Report describes & classifies system anomalies during testing.

# Test Incident Reporting – Sink Example

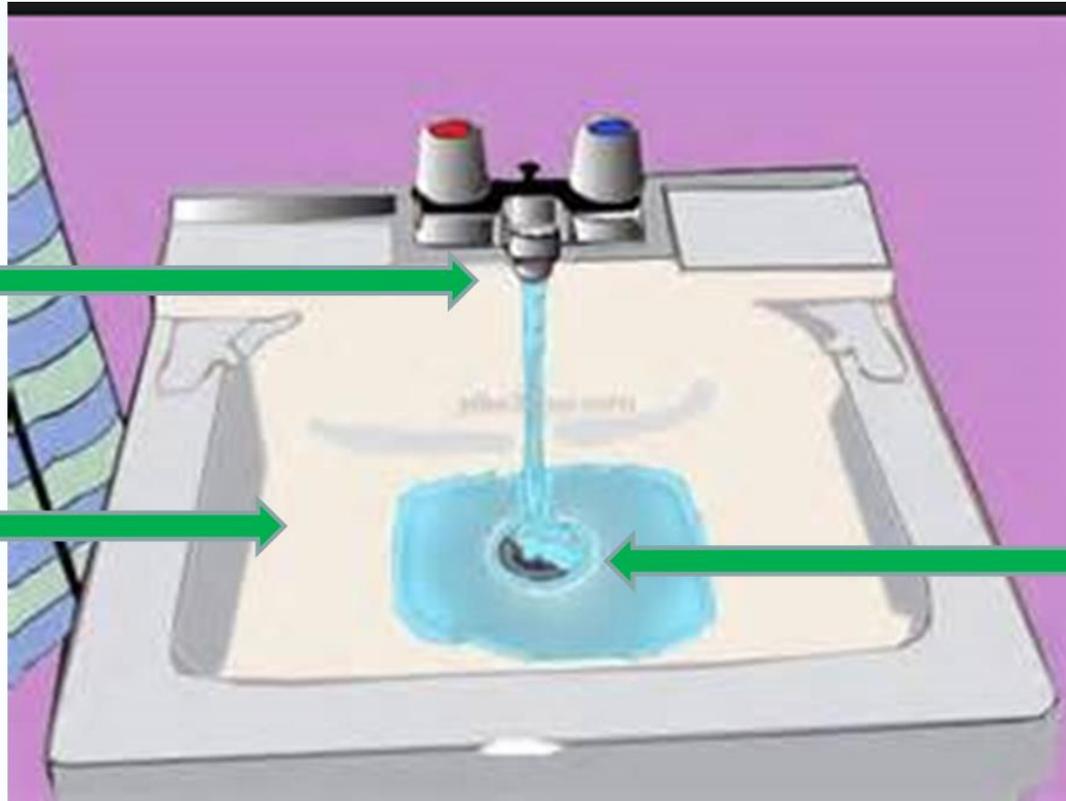
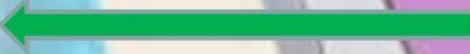
**Faucet Spout**  
**(New Defects)**



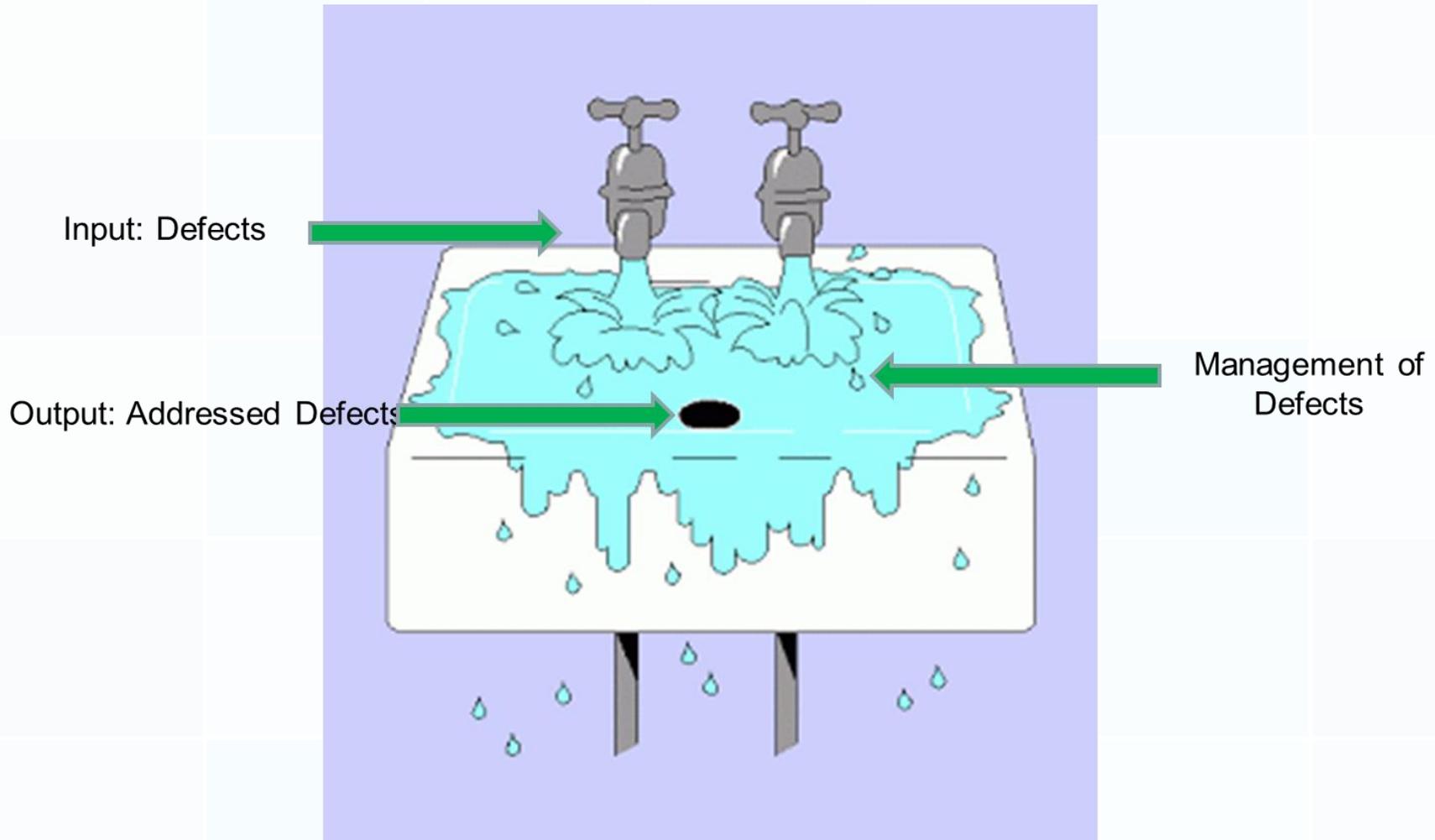
**Sink Basin**  
**(Collection of Defects Not Completed)**



**Drain**  
**(Defects Completed)**



# Test Incident Reporting - Sink Example - Continue



# Test Incident Reporting Pitfalls

- Incidents do not trace to a test case.
  - Result: Incident may be categorized as an enhancement rather than a defect.
- Defects are either ignored or minimized (i.e., overflowing sink).
  - Result: This could result is users losing confidence in the quality of the system. Also, this could lead to an exorbitant cost to fix the defect at a later stage.

# Test Incident Reporting Tips

- Ensure that all defects are associated from a test procedure to test case to requirement.
- Timely management of the incidents (i.e., sink).
  - Consistent categorization of defect severity and impact levels and the priorities of what really needs to be fixed (i.e., overflowing sink).
  - Capture and monitor metrics such as: number of open/closed defects, duration of time to fix a defect, defect severity and impact by business domain.
  - Use change control process for managing system changes (e.g., enhancements).

# Test Exit Criteria Evaluation and Reporting Activities

- Assessment of test execution results against test objectives.
- This activity is important for the following reasons:
  - Provides test result visibility(e.g., trends).
  - Provides feedback on how the test work is going and allows test improvement opportunities.
  - Measures test status, test coverage, and test item against the exit criteria to know when testing is done.
  - Evaluates Exit Criteria.
- Performed in conjunction with Test Execution.
- Test Summary Report is created in this activity.

# Test Exit Criteria Evaluation & Reporting Pitfalls/Tips

## ■ Pitfalls

- Exit Criteria waived & consequences not considered or risk mitigated.
  - Result: Quality issues with the product.
- Inconsistent collection of data to determine the progress of testing.
  - Result: Inaccurate depiction of where testing is. Difficult to know how much testing is left & when we're done.

## ■ Tips

- For deviations from baselined test plan, a risk analysis, business impact, & mitigation strategy needs to be done.
- Periodically revisit the process and method for identifying and collecting the data to measure test progress.

# Test Completion Activities

- Activity that ensures the test assets are available for later use, the environments are left in a satisfactory condition, and the test results are recorded and communicated to relevant stakeholders.
- Activity occur at the end of the test project, test level, or major test activity.
- Current test assets are produced in this activity.

# Test Completion Pitfalls

- Test Completion is not planned.
  - Staff prematurely leave project & leads to project delays.
  - Limited knowledge transfer occurs. Production staff may not adequately be trained to support the production systems. Business operations may be impacted.
- Lessons learned are not captured.
  - Result: Reduce opportunity for test continual improvement (e.g., planning for next test cycle & SDLC).
- Test artifacts are not updated and archived.
  - Result: Challenges with M/O in timely addressing corrective, adaptive, and system maintenance activities. Business operations may be impacted.

# Test Completion Tips

- **Contract language includes Test Completion Activities.**
  - **Examples: (1) knowledge transfer, (2) lessons learned & corrective actions, (3) training, (4) updated test artifacts baselined & archived under configuration management, (5) test & system documentation (e.g., standard operating procedures, manual).**
- **Contracted/state project staff skilled to provide test completion activities.**
  - **Sign-off on completion of test completion activities (e.g., demonstrated proof that the production support team has sufficient knowledge to support system (within SLA and OLA) such as running batch jobs).**
- **Process in place for managing changes to the test artifacts.**

# Summary - Key Take Away

- Test planning, preparation, & evaluation are planned early.
- Test processes are in place & followed.
- Traceability is key. Requirements & test cases verify the requirements.
- State involved in all test phases.
- Test Plan is managed throughout the project.
- Roles and responsibilities (vendor/state) are understood.
- Change control process is followed for managing change.
- Release/configuration management process is followed.
- Defect are managed efficiently.

# Objectives Covered



**Understand the purpose and importance of testing.**



**Obtain high level knowledge of the Test Process Activities.**



**Learn common pitfalls and tips for addressing test challenges.**

# Consulting and Planning Division

## ■ Resources

### ■ CPD's Home page

<http://www.cio.ca.gov/cpd/default.asp>

### ■ Consulting landing page

<http://www.cio.ca.gov/cpd/consulting.asp>

### ■ CPD Project Plans and Tools

<http://www.cio.ca.gov/cpd/plansandtools.asp>



# Questions



## ■ Contact

■ [Janet.Fong@state.ca.gov](mailto:Janet.Fong@state.ca.gov)

IT Project Advisor - Testing

(916) 228-6653

■ [Andy.Wertin@state.ca.gov](mailto:Andy.Wertin@state.ca.gov)

IT Project Advisor - Governance, Risk and Issue

Management, and Organizational Change Management

(916) 403-9633