

Software Testing Workshop

STC 45th Annual Conference
Anaheim, CA

Charles Fisher
Tracey Chiricosta
Tom Witherspoon

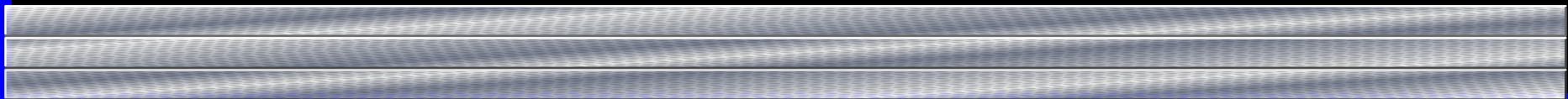
Today's Presentation

- Introduction
- Before you begin
- Writing a test plan
- Testing the software
- Tracking problems
- Software assessment
- Wrap-up and review



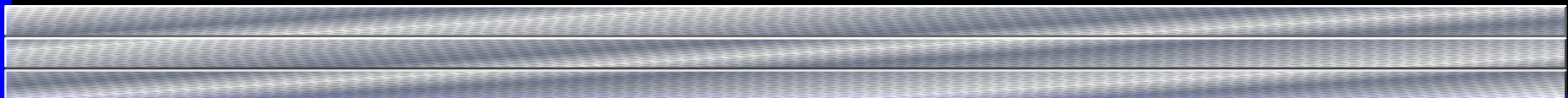
Before You Begin

- Can't test a program completely.
- The purpose of testing is to find problems in programs, not verify they work correctly.
- Effective testing is a process of “creative destruction.”
- Testing is an essential part of software development.



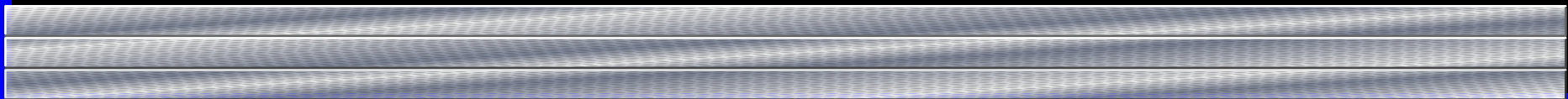
Before You Begin

- Define testing objectives:
 - Meeting system requirements
 - Finding bugs in programs
 - Confirming user acceptance
 - Finding limits of performance
 - Verifying documentation



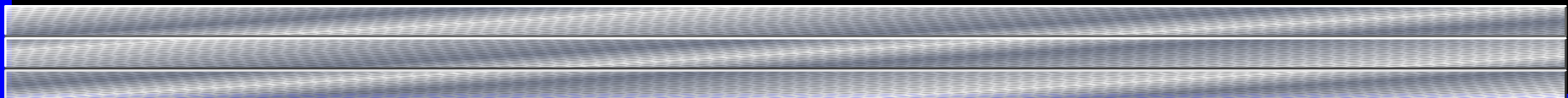
Before You Begin

- Work within resource, project, budget, and schedule limits.
- Coordinate testing schedule with team members.
- Document these decisions.
- Review documented decisions.



Types of Testing

- Unit testing
- Integration testing
- Functional testing
- Regression testing
- Acceptance testing



Testing Methodology

Phase	Designer Role	Tester Role
Analysis	determine feasibility and specify requirements	plan and set test objectives and test requirements
Design	specify general and detailed system design	specify tests to be developed
Implementation	code system	construct or acquire test procedures and cases
Execution	install and debug system	run and re-run tests
Maintenance	enhance and modify system	save and update tests as software changes



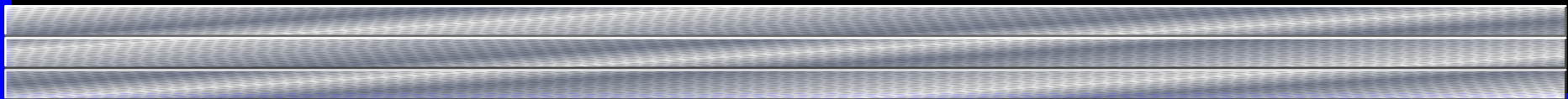
Writing a Test Plan

- A test plan should include the following sections:
 - software description
 - test description
 - test locations and participants
 - test requirements
 - references
 - problem report form
 - testing strategy and tips



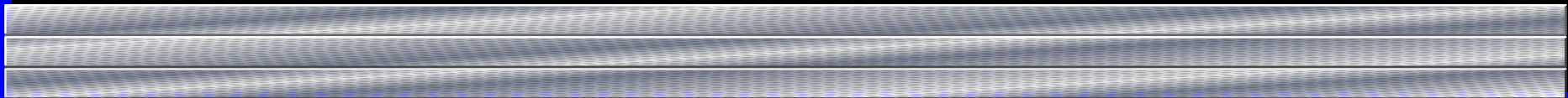
Writing a Test Plan

- Include a comprehensive outline of all elements that will be covered by the test.
- Include instructions for the testers to follow.
- Include instructions for using problem report forms or tracking systems.



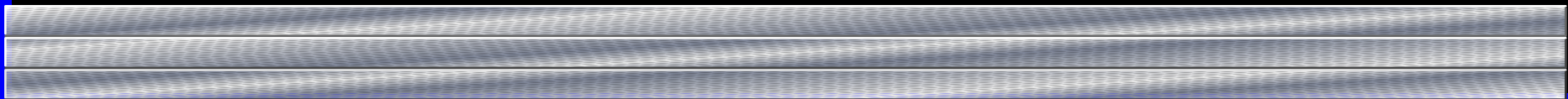
Exercise

- Choose a type of testing to conduct for your assigned system.
- List the objectives for this test.
- List a few features of the system for which you want to develop test scenarios.



“Not everything that counts can
be counted, and not everything
that can be counted counts.”

--Einstein



Testing the Software

- Make sure that the system is bulletproof or forgiving of mistakes.
- Vary your behavior and omit steps at the critical (problem) step.
- Check previous versions to isolate problems.
- Back up data before duplicating a problem.
- Watch out for “initial state” bugs.

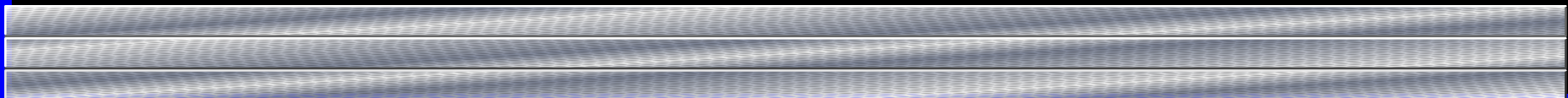
Developing and Using Scenarios

- The scenario should have a reasonable probability of catching an error. How might the program fail?
- Think backwards. If the program could fail in this way, how would you catch it?
- Scenarios should not be redundant. Choose the most effective scenario.
- Scenarios should be neither too simple nor too complex.



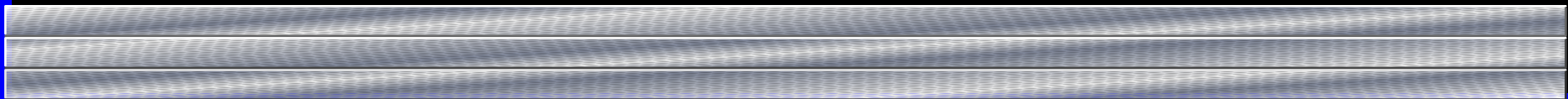
Scenario Style and Format

- Use active voice.
- Write test conditions concisely.
- Use pass/fail checkboxes to record test results.



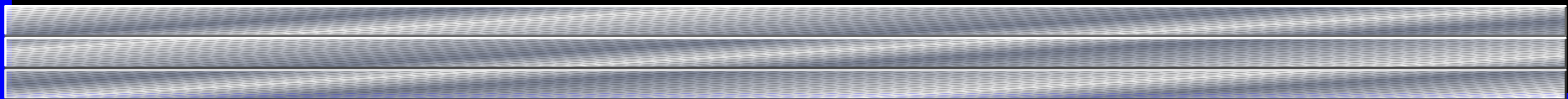
Exercise

- Write one or more test scenarios from your list of items to test.



Testing the Software

- Use only one method of reporting problems.
- Have a single repository for all problem reports.
- Every person who uses the software should report problems.
- Every problem should be reported.



Tracking Problems

- Why track problems with the software?
 - to identify problem areas
 - to collect information to re-create and correct problems
 - to control “feature creep”
 - to help measure reliability
 - to monitor and manage the testing and development process



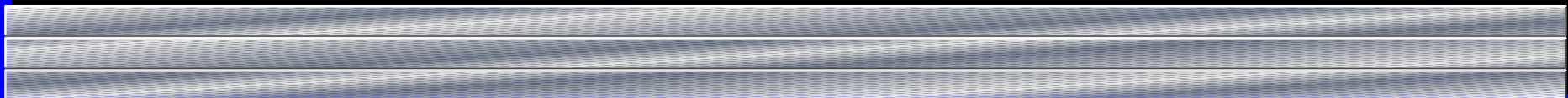
What Should You Track?

- Essentials:

- description of problem
- area of software affected
- status of problem

- Extras:

- type of problem
- priority/severity
- version/platform information
- resources assigned



Essential Information

Problem	Screen(s)	Status
Cannot add a new record	Label Application screen	New
Clone feature does not copy Status field	Correspondence screen	Open
Cannot search by ID number	Application search screen	Closed



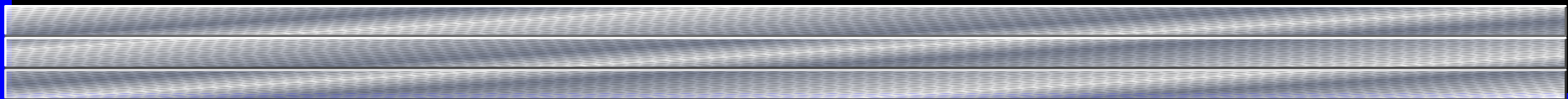
Detailed Information

Problem	Cannot add new record
Description	When I access the label application screen and click Add, I get the following error message: "Add mode not allowed."
Screen(s)	Label Application Screen
Status	New
Type	Design Issue
Priority	1
Version	1.0, 12/12/96 beta
Assigned to	Joe Smith



Description of the Problem

- Start with a one-line summary.
- Provide enough detail to re-create the problem.
- Explain *why* the problem is a problem, if needed.
- Describe all error messages.
- Attach supplemental information.
- Write problem reports immediately.



Type and Status of Problems

● Problem Types

- design issue
- bug
- documentation
- hardware issue
- database issue
- enhancement

● Problem Statuses

- new
- open
- closed/withdrawn
- fixed
- deferred



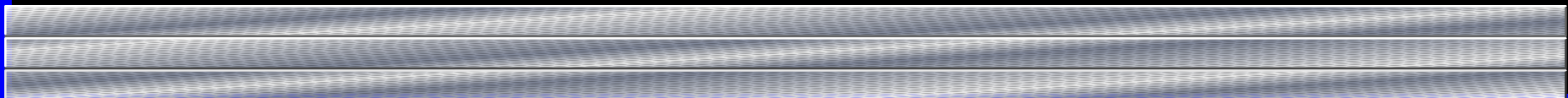
Severity of Problems

- Severity can be simple or complex:
 - **annoying:** workflow or standards issues
 - **minor:** standards issues or error messages
 - **serious:** error message or data corruption
 - **fatal:** data corruption or system failure
- Use severity to prioritize error resolution.



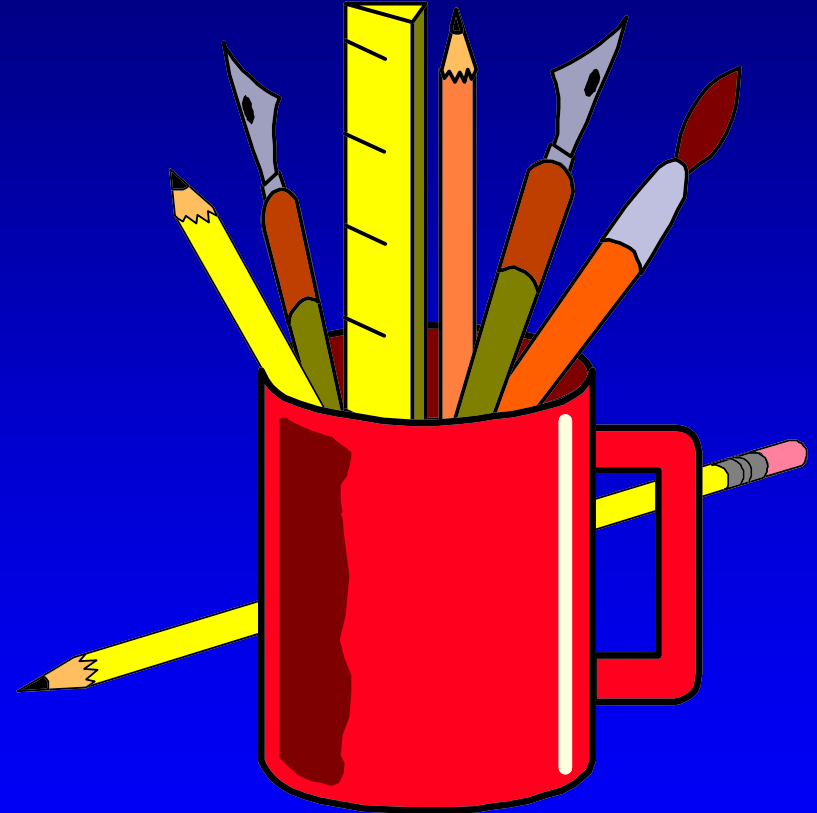
Reporting the Status of Problems

- Generate weekly status reports that show
 - total outstanding issues
 - new issues by severity
 - closed issues
 - comparison to last report
- Use reports to track status of testing and evaluate priorities.



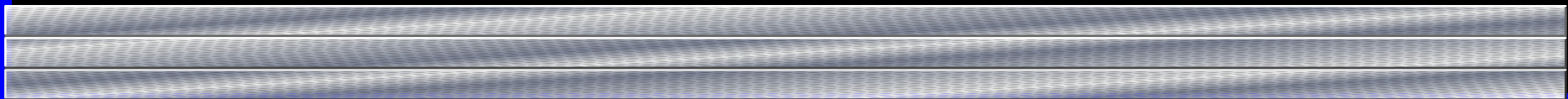
Exercise

- Using the sample problem report forms, write a few problem reports from the information your testing team has provided to you.



Assessment

- Evaluate the overall condition of the system:
 - Have testing objectives been met?
 - Have bugs and problems been fixed?
 - Once fixed, has the system been re-tested?
- Continue improving the test process.



Wrap-Up and Review

- Questions?
- For More Information...

