Data-Oriented Quality Solutions
BEST PRACTICE
GUIDELINE SERIES

Planning Test Strategies

Introduction

A test strategy maps techniques for testing against the various types of verification and validation test types available and attempts to prioritize how testing resources will be allocated generally across a project. Each individual situation is ultimately evaluated on a case-by-case basis, but the general strategy documents the perceived risk areas being addressed by the test plan.

Test Techniques

- **REQUIREMENTS TESTING -** Assures that the system performs as specified in order to meet the needs of the users. The specifications map back to specific requirements and those requirements represent the real and complete user set of requirements. All policies and regulations have been adhered to. Requirements testing looks for complete traceability of each discrete requirement statement to each detail unit of technology implementation and documentation.
- **REGRESSION TESTING -** Assures that anything unchanged still performs correctly. All unchanged system components still function as expected and all manual procedures surrounding the systems are still correct. The potential scope of testing is theoretically largest in regression testing since the entire universe of technology and documentation outside of a project's scope should be retested to assure no negative unplanned impacts. In fact, risk assessment can be used to dramatically limit the scope of regression testing to those technology and documentation components that are reasonably at risk.
- **ERROR HANDLING TESTING -** Assures that errors can be prevented or detected, and then corrected. A variety of errors of different severity are intentionally introduced into the test cycles.
- **MANUAL SUPPORT TESTING -** Assures the people-computer combination produces the right result. All manual procedures required to use and operate the system functions have been developed and implemented.
- **INTERSYSTEMS TESTING -** Assures that data is correctly passed from system to system. All intersystem parameters have changed and foreign system documentation has been updated.
- **CONTROL TESTING -** Assures that controls reduce system risk to an acceptable level. All data and process reconciliation procedures work with all appropriate manual controls in place.
- **PARALLEL TESTING -** Old systems and new system are run and the results are compared to detect unplanned differences. The old and new system outputs can be reconciled within the limits imposed by new system functions.
- **STRESS TESTING -** Assures that the system performs with expected high volumes. Sufficient technical resources have been allocated so that operational capability can be maintained under maximum volume and pressure.
- **EXECUTION TESTING -** Assures that the system achieves desired level of performance. All transaction turnaround time is adequate and execution doesn't preclude other operational activities within the technical environment.
- **RECOVERY TESTING -** Assures that the system can be returned to an operational status after a failure. Backup data and recovery procedures are in place and working, including major disaster recovery planning and rehearsal.
- **OPERATIONS TESTING -** Assures that the system can be executed in a normal operational status. Only production documentation is needed in order to operate system functions.
- **COMPLIANCE TESTING -** Assures that the system is developed in accordance with standards and procedures. All standards have been followed and all documentation is complete and in place.
- **SECURITY TESTING -** Assures that the system is protected in accordance with importance to organization.

Test Types

There are five basic types of testing available to each project:

- **TEST DATA & SCRIPTS -** Test sequences, conditions, records, and transactions created solely for testing.
- **PRODUCTION DATA -** Actual transactions that have been previously recorded to be used for testing purposes. Business transactions may need to be rearranged to meet the new formats and needs of the new system.
- **SIMULATION / PROTOTYPING -** Testing prior to the actual system being developed by exercising an early version of the system to see how it will behave. The objective is to observe how portions of the system will operate and perform without actually building them.
- **REVIEW / CONFIRMATION -** An analysis of a deliverable by persons other than the authors (or by automated means) to verify that the deliverable meets standards-needs-specifications and is defect-free.
- **INSTRUMENTATION / TRACING -** Counting of events, transactions, instruction execution, or tracing the flow of events to determine which events were executed and frequency of execution.

Prioritizing Techniques & Types

The optimum combination of types will depend upon the testing objectives that have been set, the amount of time available, and the skills of the test team. The following table illustrates a typical prioritization for a large project with diverse testing objectives:

	TEST DATA & SCRIPTS	PROD. DATA	SIMULAT./ PROTO.	REVIEW / CONFIRM.	INSTRUM. / TRACING
REQUIREMENTS	High	Medium	Medium	High	Low
REGRESSION	High	High	Low	Medium	Low
ERROR HANDLING	High	Low	Low	High	Medium
MANUAL SUPPORT	High	Low	Medium	High	Low
INTERSYSTEMS	High	Low	Medium	Medium	High
CONTROL	Medium	Low	Low	High	Medium
PARALLEL	Medium	High	Low	Medium	Low
STRESS	Low	Low	High	Medium	High
EXECUTION	High	High	Low	Medium	Low
RECOVERY	Low	Low	Medium	High	Medium
OPERATIONS	High	High	Low	Medium	Low
COMPLIANCE	Low	Low	Low	High	Medium
SECURITY	High	Low	Medium	High	Medium

More detailed test planning and execution is conducted within the context and constraints of this prioritization. The above table is an example. Each project must establish its own priorities. The resulting strategy is then enacted in a test plan used throughout the project lifecycle.

Concentrate resources on where the important defects will be found!