End User Testing for Performance Qualification of GXP/ Part 11 Systems – Off The Shelf (OTS)

Dr. Teri Stokes, GXP International, Concord, MA – USA

www.GXPInternational.com
Email: GXPINTL@RCN.com
Context for Computer Validation Testing

Operating environment (Quality Control Lab)

Computerized system (HPLC LIMS System)

- Computer system
  - Hardware
  - Software

- Work Process
  - Equipment
  - People
  - SOPs

Infrastructure Platform (IT/IQ)
Software Application (User/PQ)
User Acceptance at System Go-Live

1. SYSTEM IDEA
   Needs Analysis, RFP & Contract

2. SYSTEM PLAN
   What? URS

3. DESIGN
   How? FRS & SDD
   SDLC - SW Development Life Cycle
   Test Fit to Design

4. BUILD
   Program or Configure

5. TEST
   Verify to SDD & Release
   PQ
   Test Fit to Install Specs.
   Audit OQ

6. COMMISSION
   Accept & Validate to URS

7. OPERATE
   Use & Monitor

8. MAINTAIN
   Fix & Modify

9. RETIRE
   Decommission & Replace

HPLC LIMS Application Life Cycle

HPLC LIMS User Group

HPLC LIMS Platform

1. Configure
   Test Fit to Install Specs.

2. Install

3. Test

OQ – Operational Qualification
IQ – Installation Qualification
PQ – Performance Qualification

Slide 3
Change Control and Ongoing Testing

Application Life Cycle

1. SYSTEM IDEA
   Needs Analysis, RFP & Contract

2. SYSTEM PLAN
   What? URS

3. DESIGN
   How? SDD

4. BUILD
   Program or Configure

5. TEST
   Verify to SDD & Release

6. COMMISSION
   Accept & Validate to URS

7. OPERATE
   Use & Monitor

8. MAINTAIN
   Fix & Modify

9. RETIRE
   Decommission & Replace

System User

Software Supplier

SDLC - Software Development Life Cycle

Platform Systems

Change Symbol

Slide 4
Application User’s PQ* Package

Prepared and maintained by User Department(s) Team

Standard user’s CSV package

QMS

Validation Plan

Test Plan(s)

Needs Analysis, RFP, Contract, URS, SLAs

Startup & Ongoing

Change Control Log, QA Audit Log, Supplier Reports & BDG Minutes

Test Cases, Scripts, Data & Result Logs

Trace

Test Summary Report(s)

User Manuals, CVs, Trng Records, Dept. SOPs, WIs, Help Logs

Users’ PQ Package Summary Report

*PQ = Performance Qualification: Application meets URS workflows
CSV Package Team Model

- **Package Sponsor**
  - Funds and Approves Validation Work

- **Team Leader (Project Mgr.)**

- **Package Manager (QC of Pkge. Docs.)**

- **Ad hoc Members (Tester, Witness...)**

- **Test Coordinator (Test Plan, Scripts...)**

- **Quality Assurance**
  - Trains Team and Audits CSV Pkg.

**CSV Package Team**

Team to Develop and Maintain a CSV Pkg.
Requirements are required for testing.

All formal testing requires a metric or benchmark (e.g. requirement) to test against. If you don’t know your “expected results” then testing is meaningless and validation is not possible.
What do you need in a system?

IQURS

Identify infrastructure needs

PQURS

Identify work process needs

Requirements
Do You Have a Requirement or a Wish?

Can it be tested?

Yes = Requirement

NO = Wish
# Step #1 Develop User Requirements

<table>
<thead>
<tr>
<th>Req. ID</th>
<th>User’s Work Step</th>
<th>System Role – User Req.</th>
<th>User Accept. Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>UR01</td>
<td>Secure data entry</td>
<td>Control user access and log off intruders</td>
<td>Use pos. and neg. IDs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Log off after 3 failed tries</td>
</tr>
<tr>
<td>UR02</td>
<td>Track data edits</td>
<td>Audit data changes – Part 11</td>
<td>Change items. Check screen. Run Report.</td>
</tr>
</tbody>
</table>
### Production Planner:

<table>
<thead>
<tr>
<th>UR ID</th>
<th>Work Process Steps</th>
<th>User PQ Tests</th>
<th>Challenges</th>
</tr>
</thead>
<tbody>
<tr>
<td>UR016</td>
<td>Picking list to production orders</td>
<td>Check signals for creating a picking list</td>
<td>What happens if material is physically non-existent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check principals for picking a batch</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check principals for picking a material</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check rest or shelf life</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check authorized status</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check special dosage</td>
<td></td>
</tr>
</tbody>
</table>
### What Does the User’s Process Need? (URS)

#### Production Operator -1:

<table>
<thead>
<tr>
<th>UR ID</th>
<th>Work Process Steps</th>
<th>User PQ Tests</th>
<th>Challenges</th>
</tr>
</thead>
<tbody>
<tr>
<td>UR017</td>
<td>Material is picked and taken to working station for weighing of components</td>
<td>Check printing of labels (if weighing station)</td>
<td>Check correct storage bin</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check feedback rest material to warehouse</td>
<td>Handling of wreckage</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check reg. of loss (waste)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check special dosage</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Finance Checker: Check product costing)</td>
<td></td>
</tr>
<tr>
<td>UR018</td>
<td>Feedback per workstation - time and materials</td>
<td>Check function in system</td>
<td>Check rules for yield</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Finance Checker: Check signals to cost price)</td>
<td>Creation of extra batches</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Finance Checker: Check product costing)</td>
<td></td>
</tr>
</tbody>
</table>
Data Capture – Lab URS View for PQ

- Lab Methods
- Instruments
-Capture & Analysis Systems (PCs)
- Analysts & Supervisors
- Reagents
-SOPs & Work Instructions
-Sample Handling

Slide 13
OECD GLP – Acceptance Testing

• “…there should be evidence that the system was adequately tested for conformance with acceptance criteria … prior to being put into routine use.”

• “Formal acceptance testing requires the conduct of tests following a pre-defined plan and retention of all testing procedures, test data, test results, a formal summary of testing, and a record of formal acceptance.”
Step #2 – Map System to Work Process

1. Team agrees a common workflow for use of system in the work process (URS).

2. Workflow steps interacting with the system are identified.

3. Critical workflow steps are described in an SOP or Work Instruction.
4. Each workflow step has positive and negative challenges identified in testing step procedures.

5. Test scripts are organized by logical groups of workflow activities into Test Cases.

6. Test Scripts and related documents are managed with large envelopes.

Slide 16
Step #3 – Make Logical Groups of Test Scripts

Don’t get tangled up in lots of Test Scripts! Organize them into Test Cases based on work process flow and requirements.
Step #3 – Make Logical Groups – Test Cases

Outline adapted from - IEEE Std. 829-1983

1. **Test Case Identifier** – unique & related to Test Plan ID
2. **Test items** - scope of features being tested & URS items addressed with table of Test Scripts to be used
3. **Input requirements** - user roles, system privileges, input data types ...
4. **Output requirements** - reports, listings, screen printouts
5. **Environmental needs** - user materials, training, physical setup
6. **Special procedure requirements** - anomaly handling
7. **Inter-case dependencies** - cases to be run prior to this
### User Test Case Descriptions (PQ)

To include normal, problem, and stress conditions in user’s work process environment.

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TC01</td>
<td>Work Area Preparedness – SOP, WI, Manual</td>
</tr>
<tr>
<td>TC02</td>
<td>System Setup/Admin – User Profiles, DB schema</td>
</tr>
<tr>
<td>TC03</td>
<td>Work Process – Activity A (<em>vanilla run</em>)</td>
</tr>
<tr>
<td>TC04</td>
<td>Work Process – Activity B (<em>chocolate issues</em>)</td>
</tr>
<tr>
<td>TC05</td>
<td>Work Process – Activity C (<em>strawberry stresses</em>)</td>
</tr>
<tr>
<td>TC06</td>
<td>Special Challenges – Multi-user, Problem work</td>
</tr>
</tbody>
</table>
Server Test Case Descriptions (IQ)

To include normal, problem, and stress conditions in IT/IS environment.

- TC01. Hardware Operations
- TC02. Operating System (O/S) & Software Tools
- TC03. Database Engine (DB) & Query Tools
- TC04. Network Operations – LAN/WAN
- TC05. Platform Routine Backup & Recovery
- TC06. Platform Disaster Recovery – System & Data
Test Step # 4 Develop a Test Plan

Prepared and maintained by User Department(s) Team

Standard user’s CSV package

Test Plan(s)
Startup & Ongoing

Test Cases, Scripts, Data & Result Logs

Test Summary Report

URS

Trace

System Control

Human Control

Slide 21
Test Step # 4a Develop a Trace Matrix

No testing project is too big or too small for a trace matrix. Always trace Test Scripts to Requirements to ensure that all requirements have been sufficiently tested.
IEEE Format for a Software Test Plan

Software Test Plan Outline - IEEE Std. 829-1983

1. Test plan identifier
2. Introduction
3. Test items
4. Features to be tested
5. Features not to be tested
6. Approach
7. Item pass/fail criteria
8. Suspension criteria & resumption requirements
Software Test Plan Outline - IEEE Std. 829-1983

9. Test deliverables
10. Testing tasks
11. Environmental needs
12. Responsibilities
13. Staffing and training needs
14. Schedule
15. Risks and contingencies
16. Approvals

(IEEE Tel: 800-678-4333)
Who Is Who in Testing?

Test Coordinator – writes test plan, test cases, and prepares testing materials. Trains testers & witnesses, manages testing process, and tracks results and anomalies. Writes conclusion per case. Writes test summary report. Organizes test records in a test records binder.
Who Is Who in Testing?

**Script author** - can never be tester or approver of same script, but can witness testing of same script

**Script approver** – independent reviewer of script for content and strategy

**Tester**- performs test, records system response, and signs script for accurate recording of system response

**Witness** - verifies that tester is prepared and ready to perform test, checks presence of final test records, and signs script for compliance to testing practices SOP

Roles & Responsibilities
1. **Keep Tester Alert** – author scripts that can be executed within 30-50 minutes.

2. **Rule of Three** – test key functions 3 times using a variety of challenges – normal, problem a, and issue b

3. **Trace to Requirements** – identify requirements being addressed by each script.

4. **System Output** – have at least one screen print or report per test script. Identify label info needed for output.

5. **More than Pass/Fail** – have tester write out an observed system response.
5.1 **System requirements Specifications**: Established, documented end user requirements are essential. (GXP, Part 11, work process – scanning, scalability, operating environment requirements)

5.2 **Documentation of Validation Activity**: Management approved plan(s), procedures and report(s).

5.3 **Equipment Installation**: Qualify installation prior to testing.
5.4 Dynamic Testing: 5.4.1 - Test conditions, simulation tests, and live, user-site tests. 5.4.2 - Software testing to include structural, functional, and program build testing. 5.4.3 – Quantifiable tests recorded in quantified, not pass/fail terms.

5.5 Static Verification Techniques: Document and code inspections, walk-throughs, and technical reviews.

5.6 Extent of Validation: System risk to product safety, efficacy, and quality. System risk to data integrity, authenticity, and confidentiality. System complexity.
5.7 Independence of Review: CSV to be performed by persons other than system developers.

5.8 Change Control (Configuration Management): Systems to control changes and evaluate extent of revalidation needed. Regression testing is a key tool.

5.6 Extent of Validation: Based on

- System risk to product safety, efficacy, and quality
- System risk to data integrity, authenticity, and confidentiality
- System complexity.
Step #5 - Prepare Test Script Materials

Prepared by Test Coordinator & Package Team

- Standard Test Script Envelope
- Test Data Form, SOP, Work Instr.
- Incident Form (Red Note) & WI
- Test Script
  - URS Trace & Script Pass/Fail T/W Signatures
- Step Procedures
  - Result Logs & System Issues
- Witness Checklist
- Support Materials, Result Printouts...

Large Mailing Envelope
Step #6 - To Do’s in Formal Testing

1. Check script, logs, data, & testing area.
2. Record all system responses as they occur.
3. Use only indelible ink to record results.
4. Correct with single line through record & new result next to it. Initial, date & explain.
5. Draw single line through unused log spaces.
6. Label, sign & date all printouts or CD/diskettes.
Step #6 - The Don’ts in Formal Testing

1. Don’t use pencil or other erasable media.
2. Don’t correct by using white-out or scribble over to obliterate prior result.
3. Don’t use check marks, dittos, Y/N or other abbreviations. Write results & comments in full.
4. Don’t leave large blank spaces in result logs. Line through.
5. Don’t forget to sign & date all output documents & logs.
Witness Participation in Testing

1. Checks contents of T.S. Envelope with Tester for completeness and understanding.

2. Checks System Setup with Tester for logon access. Watches start of first testing step.

3. After testing, T.S. Envelope is re-examined. Checks Test Script and printouts for completeness, signatures, dates and labels.
Step #7 – Prepare a Test Summary Report

Testing is finished at last!!
Reporting on Test Results

Prepared and maintained by Site User Team

- **Standard user’s CSV package**
- **Test Plan(s)**
- **URS**
- **Trace**
- **Test Summary Report**
- **System Control**
- **Human Control**

Slide 36
1. **Test Summary Report Identifier** - Unique ID traceable to associated Test Plan.

2. **Summary** - Describes items tested (application version), test environment (platform system), and test documentation used (Test Cases, Test Scripts, T.S. Envelope contents).

3. **Variance** - States any deviations from Test Plan or Test Scripts and reason why.

4. **Comprehensive Assessment** - Discusses assumptions and limits to scope of testing. Were scope of testing and results obtained sufficient to assess system reliability? Discuss reasons for limits chosen.
5. Summary of Results - Gives table of testing results per Test Case. Table of anomalies and their resolutions. List of outstanding issues and risks (unresolved anomalies).

6. Evaluation - Pass/Fail conclusion based on test results and criteria in the Test Plan.

7. Summary of Activities - Tester/Witness staffing, task list from Test Plan with updated status.

8. Approvals - Names, titles, signatures, dates and meaning of signatures.

9. Appendix - Table of Contents list for test documentation
Step #7 - Store Test Results for Audit

Prepared by Test Coordinator at Site

Site Test Binder

SOP for Testing

T/W Training Records

T. Case 1 - SOP & Document Review

Test scripts, test data & Result Logs


Site Test Summary Report

Site Test Binder

T. Case 2 - System Admin. Workflow

Step Step Step

Test scripts, test data & Result Logs


Site A

T. Case 3 - Work Process Workflow(s)

Step Step Step

Test scripts, test data & Result Logs

Over time systems change. An Ongoing Test Plan sets up the process for maintaining a validated status as change occurs. It identifies major, moderate, and minor types of system change and corresponding degree of formal testing to be performed any validation tasks to be updated.
Audit Points for Test Documentation

• Test Plan was approved with or after Validation Plan approval date and before Test Script approval dates.

• Test Plan describes system to be tested, gives specific strategy for testing, and identifies tasks and roles responsible.

• All issues arising are recorded, tracked, and resolved.

• Repeat testing is performed using a new copy of a script and the run number is identified.

• Test Script author is not tester or approver for same script.
Audit Points for Test Documentation

- All printouts are labelled with tester ID, date, and script ID.
- All log entries are made in indelible blue or black ink.
- No abbreviations (P/F, Y/N), ditto or check marks were used.
- Signature page identifies names, initials, signatures, and Tester/Witness roles.
- Test Script identifies Requirements being tested, Testing Site, Test Run, Author, Approver, Tester, Witness and overall conclusion.
Audit Points for Test Documentation

- Test Summary Report clearly describes what happened, how problems were handled, who was responsible, and how the Test Plan was followed or what deviations were made and why.

- Test Summary Report should show that test execution was consistent with Test Plan strategy.

- PQ testing uses GXP work process SOPs and work instructions to test their suitability for working with the system.

- IQ testing uses IT Dept. SOPs and work instructions to test their suitability.
## Formal Testing Major Themes

<table>
<thead>
<tr>
<th>MANAGEMENT CONTROL</th>
<th>SYSTEM RELIABILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Formal Testing Practices SOP</td>
<td>• Requirements &amp; Specifications</td>
</tr>
<tr>
<td>• Approved Test Plans, Test Scripts, &amp; Test Summary Reports</td>
<td>• Trace Matrix – Tests to Requirements</td>
</tr>
<tr>
<td>• Anomaly Tracking Process</td>
<td>• Defined Acceptance Criteria in Test Procedure</td>
</tr>
<tr>
<td>• User &amp; Support Documents</td>
<td>• Limits, Logic &amp; Problem Testing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DATA INTEGRITY &amp; PRIVACY</th>
<th>AUDITABLE QUALITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Tester &amp; Witness Signatures</td>
<td>• Approved Plans &amp; Reports</td>
</tr>
<tr>
<td>• Result Logs in Indelible Ink</td>
<td>• Traceable test coverage for all Requirements &amp; Specifications</td>
</tr>
<tr>
<td>• IQ for Automated Testing Tools</td>
<td>• Security &amp; Disaster Recovery Tests</td>
</tr>
<tr>
<td>• Security &amp; Disaster Recovery Tests</td>
<td>• Audit Trail Tests</td>
</tr>
<tr>
<td>• Audit Trail Tests</td>
<td>• Documented Anomaly Resolution</td>
</tr>
<tr>
<td></td>
<td>• Independence of Testing Process</td>
</tr>
</tbody>
</table>

Slide 44
Thank You!

Merci
Tak, Tack, Takk
Gracias
Obrigado
Spasibo
Nandri
Cobjai

It’s the little things that can bite you in an audit of test records!

Common Sense Computer Validation
How would you answer these questions?

The software supplier has tested this software a lot and other people have bought it and are using it, so why do we have to validate it with our own testing?

Why test off the shelf systems that everyone else is using in their businesses?

Why must end users be involved in the PQ testing?

Can’t IT or outside testing folks do the job for us?

Our job is our work process, how should we know what to do for testing a computer system?
Any Questions or Comments?