

Success through Test Strategy @ NIIT

Neeti Churamani
Kumud Iyer

Abstract

Every software project is different in nature and a lot of its success depends on the type and kind of verification validation.

- *We have tried to capture some of the challenges faced by the team, which was doing a field remediation project. This project was significantly different from Y2K projects handled, as in the latter; generally the existing field was expanded in-place. The difference in this project was: The impacted fields were added as new entities in existing Datasets. Moreover, Y2K projects were only for date field expansion, whereas this project was on five different business entities and related business logic.*
- *The data carried forward in the newly created fields was consistent with the legacy values. This was handled by changing the processing in the impacted routines and by creating Bridge routines for first time input datasets.*
- *A special treatment logic was created for the cases where legacy field value limits were exceeded.*

We have tried to bring out the experiences, learning and in process improvement in this project lifecycle. This paper focuses on sharing a success story of field remediation project, through effective Test Strategy and benefiting the community with NIIT's experience.

Background

The client, a leading Media Research Company, is a major provider of television ratings and audience estimates information services, serving national and local customers including television networks and affiliates, independent stations, syndicates, cable networks, cable systems, advertisers and their agencies.

The broadcast television industry is in the process of shifting its method of signal transmission from analog to digital. As a result of this decision, the client evaluated their information processing systems and found the need to initiate a Field Expansion project to accommodate advances in the video broadcast industry surrounding digital transmission standards.

As digital stations begin broadcasting, they work to stay ahead of the industry by assuring the ability to measure and appropriately credit viewing.

As part of the analysis done for Digital Conversion, the number of applications that were found to be 'digitally' impacted was thirty-six and the project "Conversion for Digital Signal Analysis" was initiated.

NIIT was awarded this project in June 2002, as it showed the following strengths

- Its experience in the technology viz. PL/1, SAS, Easytrieve, Assembler on IBM mainframes
- Experience in Y2K projects
- Established Testing and LMM practices

The Challenge

The client provided a Conversion Strategy that required five business entities to be digitally impacted along with business logic. This was a critical project as it was first of its kind for NIIT (as this was different from the Y2K projects done in the past) and opened a new domain of Media Research for our organization. The applications to be remediated were in various mainframe languages like PL/1, Assembler, Easytrieve, SAS and TSO/ISPF etc.

The challenge was to test the application without having enough time to gain indepth functional knowledge of the Critical business applications. It was a highly complex system processing huge amounts of data daily, in order of about 300 million data items a day and thus had very stringent batch window requirements for the converted system.

Out of the various levels of the testing, Regression Testing was the only reliable way to ensure that the remediation process did not inadvertently introduce errors into the existing functionality within the Business Rule as we had upgraded these rules as per the Conversion Strategy.

The objective of the Testing Phase was to ensure that the software components of various impacted and non-impacted modules of the applications worked as intended after the Code Remediation

phase and hence the functionality of the integrated application was retained.

This phase was recursive and was to be repeated, as many times as there were levels of testing.

Testing Strategy

Since this project was not like typical application development/enhancement, hence the types of testing which were decided for this were:

Unit Testing:

This was done for every routine by verifying that the digital fields, which got added as a part of remediation, contained correct values. For example, for a field called station code which in legacy had a value of "9998", after remediation the corresponding digital station code value was "000009998". In case of a station code exceeding a value of 10,000 the legacy values was set to 0.

Regression Testing:

The scope of regression testing was to compare the output of integrated units (at various levels) with the base lined production runs.

An automated tool was used which scanned the JCL and PL/1 programs and generated cross-references of programs to logical files and thence to physical files.

Integration Testing:

The scope of integration testing was to test the relevant jobs, of impacted applications, to ensure that all interfaces and bridges functioned correctly even after introduction of digitally converted routines within various jobs.

The workflow of the Field Expansion project was as follows:

The project team was divided into two groups – Development team and Testing team.

The Development team was responsible for:

1. Impact Analysis of Digitally impacted routines
2. Creation of Unit and Regression Test Plans
3. Code Remediation
4. Review of Remediated Codes
5. 1st round of Unit and Regression Testing

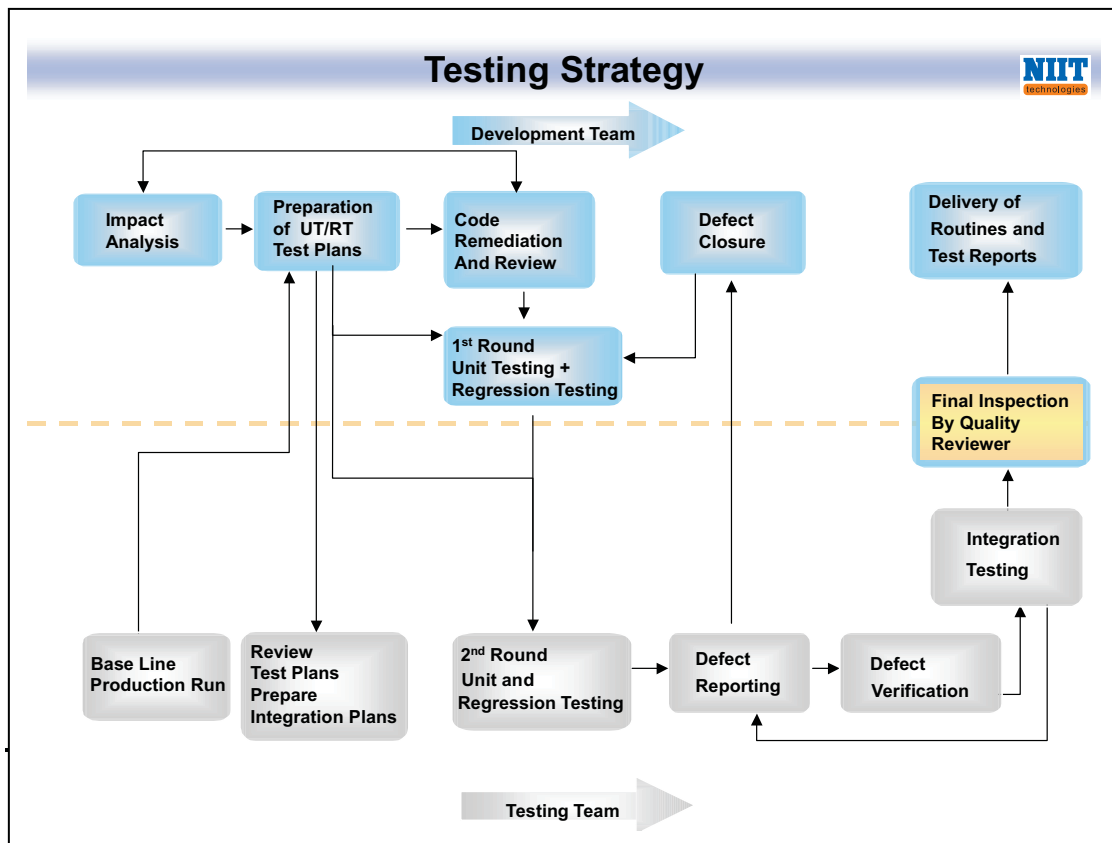


Figure 1

The Testing Team was responsible for:

1. Baseline production run
2. Review of the Unit and Regression Test Plans created by development team
3. 2nd round of Unit and Regression Testing
4. Creation of integration Test Plans
5. Integration Testing

The client did the Acceptance testing with support from the testing team.

The various steps followed during the course of the Testing process are as follows:

Unit Testing Process:

1. The developers carried out the 1st round of Unit Testing using copies of production data to ensure that the routines gave the desired output after remediation and using expanded datasets containing expanded field values. Boundary conditions of expanded fields were tested to ensure compliance to Conversion Strategy.
2. 2nd Round testing was done by the Testing team to ensure correctness.

Regression Testing Process:

1. Identify input / I-O / output files for every digitally impacted routines. The automated tool that analyzed JCL and PL/1 source codes was used in the same.
2. Baseline Production jobs containing impacted routines.
In this step, copies of Production datasets were taken and baseline runs were done of jobs and all the datasets of these jobs were preserved. These were later used for comparisons with Digital run datasets.

3. Digital run

In this step, copies of baseline run input datasets was created and the digitally converted routine was executed. The output files created were preserved with different names indicating that these were digital run outputs.

4. Comparison of Baseline run and Digital Run

In this step we carried out an intelligent comparison of the baseline files and the regression files. The Regression Testing was considered to be complete only all the output datasets of the two runs compared successfully.

Integration Testing Process:

1. Check that all the digitally impacted routines were successfully Unit and Regression tested.
2. Setup up copy of the production jobs by setting up input datasets of nonimpacted as well as expanded datasets for impacted routines.
3. Execute the new production job containing the digitally

impacted and nonimpacted routines.

Performance Testing:

This testing was conducted as part of the Baseline run and Regression Testing cycles where the CPU usage time of the production baseline was compared with that of the digital routine. In case of routines where CPU usage was found to be exceeding production runs substantially, routines were optimized to ensure that the jobs run time was retained closely equivalent to the original jobs.

One of the success criteria's for the regression testing was that the time increase for an individual program should not exceed 10% and the time increase for the overall job should not exceed 2%.

All the above-mentioned cycles of Testing were repeated till all identified problems and issues were tracked to closure.

Testing Process Improvement

One of the major process revisions that were made to the Testing Phase was the merging of the Unit and Regression Testing cycles. In Phase I, we first completed all Unit Testing, then Regression tested all those impacted routines and then Integration Testing was completed at a job level.

This posed a few concerns:

- In case of an error detected at the Regression Test level, the developer had to re-run all the Unit level test cases.
- The Testing Team had to re-execute the second round of Unit Testing.
- As the routines were clubbed during Regression Testing, it was cumbersome to detect the exact location of the error.

To resolve this, during the course of a Causal Analysis and Resolution, the project team decided that each impacted unit be regression tested independently at the unit level. This helped in cycle time reduction by catching the record discrepancies at unit level itself in the subsequent Phases (Productivity Measure – Annexure A). Over and above this one round of regression testing was also done at logical integration of units.

Critical Success Factors

Use of Independent Testing Team

The testing team was responsible for the review of the Unit and Regression Test Plans, creation of Integration Test plans and to ensure vigorous and impartial testing of the remediated routines and applications. Since this team was dedicated to the project for complete duration, they were fully aware of the Conversion Strategy, business domain and system requirements of the Digital

Conversion project.

Leveraging the Client's Environment

The project team worked by linking to the customer's computing environment and utilized the tools available. This approach helped ensure that the test environment of the applications was similar to Production.

Collaborative Tools

The Knowledge Portal is a comprehensive information and knowledge management package. It enables organizations maximize their efficiency by providing generative information and allied service to create, share and apply knowledge within the organization.

The use of the KP helped the offshore team communicate easily with the Onsite team as well as directly with the Client.

Regular Weekly Conference calls and Status Reports with the client help address issues/concerns in the early stages of the project life cycle.

Access to Production data for Testing

For the testing cycles carried out by the project team, the customer allowed access to production data by creating copies of those datasets. This helped the testing of the impacted applications by ensuring that all possible permutations and combinations of data were tested right from the beginning of the Testing cycle. Potential performance issues were thus detected early on at the testing stage itself rather than in later phases.

Usage of Tools

To assist the Testing team meet the challenge of ensuring Digital Expansion without affecting the application's functionality, the following tools were used:

CompareX: This is an IBM mainframe tool that was used for comparing the output files of the Legacy and Digital runs.

Strobe: This tool is helpful during Performance testing. This was used for identifying if any of the impacted units had been inadvertently affected in terms of performance after its remediation.

In-house Tools:

Load Data: This tool (a simple PL/1 program) was used to convert original datasets into digital datasets during testing. The need of this tool was felt due to:

- Voluminous datasets
- Datasets with variable record lengths. These could not be digitalized using File Aid.

Data Dictionary generator

In-house developed tools that analyzed the JCL and PL/1 source

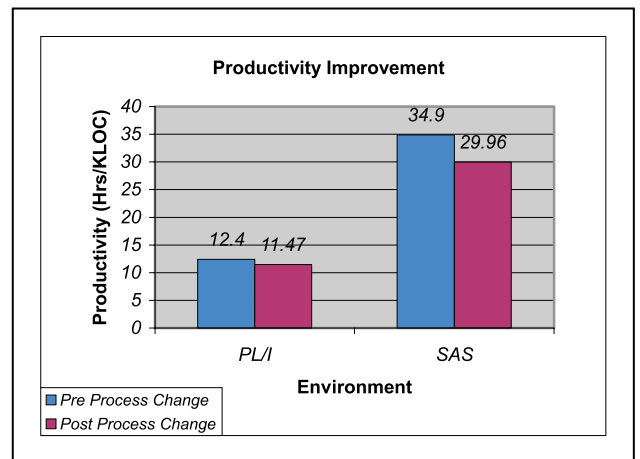
codes and created a data dictionary of programs, logical files, physical files that was useful for detecting impact, testing.

Success Measures

- 1.The project was delivered On Time.
- 2.Absolutely smooth transition to production; the applications delivered in Phase I & II are live in production while the last few applications of Phase III are slated to go live in January 2004. (Defect Measures – Annexure A)
- 3.The performance of the remediated applications in terms of CPU Time was as per the criteria specified hence ensuring that the customer didn't have to make any changes in the original job schedules.

Annexure A

Figure 2



Note: The dip in the Productivity figures (in hrs/KLOC) indicates an improved effort.

Annexure B

Figure 3

