Divercity: A case study in systems improvement through testing

Arayici, Y and Sarshar, M

INTERNATIONAL CONFERENCE ON CONSTRUCTION INFORMATION TECHNOLOGY (INCITE 2004): WORLD IT FOR DESIGN & CONSTRUCTION

LANGKAWI, MALAYSIA: 18-21 FEBRUARY 2004

Hosted by the Construction Industry Board of Malaysia, The Hong Kong Polytechnic University and The University of Salford, UK

DIVERCITY: A CASE STUDY IN SYSTEMS IMPROVEMENT THROUGH TESTING

Yusuf Arayici, Marjan Sarshar,
Construct IT, School of Construction and Property Management, University of Salford, UK

Abstract

This paper describes the case study of a system testing for DIVERCITY. DIVERCITY was an EU funded project undertaken by a European consortium of researchers and practitioners from the construction industry. They were working together in an attempt to develop VR based software that enables the industry to better undertake the client briefing, design review, and construction planning phases of a construction project. The project had the acronym DIVERCITY – 'Distributed Virtual Workspace for Enhancing Communication within the Construction Industry.

DIVERCITY's development and testing teams spread across four EU countries. That added an additional layer of complexity to the system testing. DIVERCITY used conformance approach for testing strategy to meet the user requirements of the system. This paper describes the methods used, and the lessons learned in the DIVERCITY project.

1. Introduction

This paper explores the testing methodology of the DIVERCITY project. DIVERCITY is a funded research project, which aims to develop a virtual design environment for the construction industry.

Researchers have identified the need for an integrated construction environment, which acts as a project repository, during all stages of the lifecycle (Alshawi, 1996)(Aouad et al,1997)(Issa, 1999). This aims to improve the communication between the different stakeholders and improve productivity. However, this environment has been difficult to implement. No commercial solutions currently exist. DIVERCITY aimed to develop innovative workspace technologies for the briefing and design and construction phases of the project life cycle and evaluate the results on live projects.

DIVERCITY is an abbreviation for a Distributed Virtual Workspace for Enhancing Communication within the Construction Industry. The main project goals are:

1. Creation of a **client-briefing workspace** that allows interaction and communication of design ideas between the client and the architect;

2. Creation of an interactive **design review workspace** which allows multi-disciplinary design reviews involving different stakeholders of a construction project;
3. Creation of a virtual **construction workspace** that allows the user to assess the constructability of a building, and plan and layout of the construction site;
4. Specification and development of a **software framework for integrating the above three workspaces** and sharing them over networks to support collaboration between geographically distributed project team members;
5. **Future evolution** of the virtual environment to encompass other phases of the construction life cycle, such as facilities management.

DIVERCITY's project partners are based in France, Finland, Denmark, UK and Italy. Each partner has specific areas of expertise. The Technical development teams are based in France, Finland, Italy and UK. The user teams are based in UK, Finland, Denmark and France. This team has the responsibility of developing the user requirements, as well as testing the software.

This paper explains the testing methodology of DIVERCITY, which was conformance approach (Morris, 1999), and discusses its strengths and weaknesses. It is a case study for conformance testing of systems with distributed development teams, as well as distributed user teams.

2. The DIVERCITY System

The DIVERCITY system is comprised of Communication layer, and six applications (i) Client Briefing, (ii) Lighting, (iii) Acoustic, (iv) Thermal and Heating Simulations, (v) Visual Product Chronology, lastly (vi) Site planning & Analysis, each of which handles different aspects of a construction project in a construction supply chain and how they complement each others to constitute a seamless integrated computer environment for the sake of excellence of briefing and design and construction planning.

The main strengths of the DIVERCITY system are as follows:

- Improving the co-ordination and communication between the client, design team members and construction professionals by using standard formats, and intuitive VR tools.
- Evaluating the design at the very early stages of the construction project lifecycle in terms of architectural, technical, financial and environmental aspects since VR tools allow design team to have a quick and high quality feedback on the project (Faraj & Alshawi,1999)
- Doing what-if scenarios at the detailed design stage to assess the design solution of construction project in lighting, acoustic, thermal aspects.

- Closing the gap between the construction stakeholders in design team and construction team and providing them with an integrated platform for the best buildability and applicable construction planning.
- Support stakeholders in the construction project to allow virtual collaborative spaces at geographically distant sites to work together in either synchronise or asynchronise manner.

3. Testing Methodology

Conformance and interoperability testing are two different approaches to validating software's usefulness with respect to a specification (Kindrick et al, 1996). The main difference between the two is that the traditional conformance testing approach compares an implementation against written specifications, whereas the interoperability approach compares an implementation with other implementations (Morris, 1999).

In our study, the testing approach is simply conformance testing through which the DIVERCITY system and its components are compared with the requirements specifications that have been determined by the user groups of the project team through interaction industrialist, academics and IT specialists.

Since there is no commercial implementation available in the same category with the DIVERCITY prototype system to enhance the construction project's effectiveness, interoperability approach is not fully applicable approach for this case. On the other hand, conformance approach suits well and provides some advantages. For example an advantage of conformance is that each implementation is compared to the same thing and only one thing (Morris, 1999), which is user requirements specifications. The breadth of coverage of the user requirements can be measured and redundant tests can be minimised.

However, the conformance testing approach has been criticised as it may miss tests that are important in real world implementation and it can catch things that are irrelevant in the real world (Morris, 1999). In DIVERCITY this problem was overcome through holding user-testing workshops, with industrialists, who would provide their views on the applicability of the systems to the real world.

4. Test Criteria

Throughout the testing lifecycle, the test objectives fall into six criteria:

1. Functionalities used in the applications
2. Integration for construction supply chain
3. Distribution of end users
4. Operation of the system itself
5. Usability
6. Regression testing

In each of the testing phases, the DIVERCITY system was tested against one or more of the above criteria as necessary. These criteria are defined below.

In the DIVERCITY project, testing activities were not viewed as singular activities or some individual separate tests. They were viewed as a broad workflow encompassing a continuous series of tests focused on identifying and eliminating defects and assessing product quality early and continuously throughout the development lifecycle (rational white paper, 1999) (Kruchten, 1999). For example functionality testing is conducted at every phase in an iterative manner, even though it was the main focus of the initial testing (alpha) phase. User requirements coverage was essential for the functionality testing, which were conducted as black box or specification-based testing (Whittaker, 2000).

In regards to integration for construction supply chain, construction projects need involvement of different stakeholders such as client, engineer, planner, and contractor and so on. However, they work traditionally in fragmented and also distributed environments within which there are lack of communication, lack of understanding the client needs, duplications, etc. Consequently, these factors bring about lead times and increased cost to project, inefficient use of resources, low buildable designs, and at last unsatisfied clients. In order to enhance this work conditions, DIVERCITY system provide an integrated and distributed platform upon which project stakeholders can fulfil their duties and exchange information synchronically throughout the workflow defined in the storyboard that implies the construction supply chain.

In the integration and distribution testing, the DIVERCITY applications were linked with the distribution module to provide information exchange between parties that refers to a broad but swallow form of integration. In our case in this testing category, the main goal is to demonstrate the process workflow of construction supply chain that entails the integration and distribution and how the system meets the requirements of each stakeholder perspective.

Operation of the system comprises of testing the whole domain in respect to reliability, system performance and functionality to ensure that the system fully meets the requirements, and also to close the quality gap. In the system testing, interaction between the applications was tested, which corresponds to the higher level of integration: Stakeholders were able to work on the same data at the same time. Therefore the enhanced storyboard formed the base for the system testing. The whole entire domain was run according to the enhanced storyboard, which was the full coverage of the user requirements specifications.

Usability testing was to observe the system for ease of use in terms of the user navigation and interaction, screen menus and displays, error handling,

system response, consistency and standards, user control, GUI design, documentation and help facilities. Usability testing was also conducted in each phase iteratively in order to close the quality gap in usability.

Regression testing was to measure the improvements between the testing phases that follow each other iteratively. Regression testing also covered the checking the defects encountered in the previous phases.

5. The DIVERCITY Testing Process

In the DIVERCITY, testing lifecycle incorporates three main testing phases: Alpha, Beta and Final testing. The DIVERCITY tests were also called as black-box tests. This is because, DIVERCITY tests centre on testing the programs against the written specifications and test observes the programs as black-boxes and is totally unconcerned with the internal structure of the programs (Lewis, W.E., 2000).

Furthermore, there was a symbiotic relationship between the end users and the developers in the DIVERCITY project. The entire tests in the lifecycle were conducted by the end users not the developers. The developers released the prototypes and the end users, who were distributed across Europe, tested the prototypes continuously in a collaborative manner with respect to the defined test criteria above. They continuously provided feedback to the developers throughout the test phases.

The most effective way to reduce risk is to start testing early in the development cycle and to test iteratively, with every build. With this approach, defects are removed as the features are implemented (rational white paper, 1999). Each testing phase-Alpha, Beta, Final- is conducted in an interrelated and continuous and iterative manner.

The user group employed the UML test workflow to conduct the black box tests in a stage wise manner. A typical UML workflow encompasses the stages: plan test, design test, implement test, execute test in integration tests stage, and execute tests in system test stage.

5.1. PLAN TEST

The purpose was to identify and describe the testing that will be implemented and executed. This was conducted by developing a test plan that contains the requirements for test and test strategies. In the DIVERCITY case initially a single test plan was developed, describing all the types of tests to be executed for successive prototypes at subsequent stages in the project, the test plan modified for each test stage. Based on the test plan, test efforts was measured and managed.

DESIGN TEST

The purpose was to identify, describe, and generate the test model. In the design activity, test designer analysed the target-of-test and developed the test model. Test design transformed use cases into unit, integration and

system test cases. In DIVERCITY, these use cases were situated within the storyboard derived from the user requirements for the tests. The use cases then drive the design of the software elements that implemented the tests.

5.3. IMPLEMENT TEST (ALPHA PHASE)

At this stage a definition of alpha was “a software package capable for the demonstration of the concept.” The testing of the alpha version covered primarily the building of the full-scale demonstration based on the chalet case study and the feedback obtained from the demonstration.

At the alpha-testing phase that is the initial stage of the test process, six applications (Client Briefing, Lighting Simulation, Acoustic Simulation, Thermal Simulation, 4D Simulation and Site Planning and Analysis) that form the main applications of the system were tested as stand-alone applications in UK, France, Denmark and Finland. At this stage, predetermined user requirements captured for each application were used to compare with the products and user requirements were modified as the wish list of the end users. The advantage of alpha testing is that it enabled the user groups to test the individual components and explore the possible defects belong to different applications and log them in a sheet so that it provides a better way to manage integration of all products at the integration phase.

The main objectives of the alpha phase testing were the functionality and usability testing. Each product’s conformity was observed and examined against the user requirements through functionality testing.

5.4. EXECUTE TEST IN THE INTEGRATION TEST (BETA PHASE)

At this stage a definition of beta is “a software package that can be run by the end users in a session guided by the developers.” An office building design provided by the Finish partner of the project is the case study for the beta testing. The main focus of Beta testing is the integration for exchange of information over the communication layer between the stakeholders situated in discrete places, which corresponds to the integration and distribution testing.

The storyboard was developed from the DIVERCITY use cases for integration testing, which was the process of exercising the multi-user distributed system of the DIVERCITY project. Besides, storyboard that is related to the construction supply chain enabled us to fully retest the whole functionalities of each product in order to observe the expected improvements between the alpha and beta tests in functionality as well as defects encountered at the alpha-testing phase undergoes the regression testing in which the defects have been supposedly removed (Whittaker, 2000). Furthermore, new defects discovered at the beta phase were also logged into sheets.

5.5. EXECUTE TEST IN SYSTEM TEST STAGE (FINAL PHASE)

Final phase refers to the system testing which will be a multifaceted test to assess the operation of the system in functionality, reliability, performance and usability. System testing tests the collection of the applications that constitute a deliverable system and the entire domain is considered to satisfy the criteria for a system test (Whittaker, 2000). At this stage a definition of final testing is “a software package that can be run independently by the end user”. At the final phase, the storyboard was modified from the Beta version and it required interaction of the DIVERCITY applications that refers not only broad but also deep form of integration. That is to say, the storyboard at this stage corresponds to the higher level of construction supply chain. Therefore, it entailed the interactions of the applications and the full usage of the DIVERCITY system. The final case study was again the office building used in the beta phase. Besides, the issues tested at the beta phase were retested under regression testing.

Acceptance was defined as the end user reaction to the system and their evaluations on how much the system impacts their business life. To ensure the acceptance from the end users, industrialists were involved in the integration and system testing stages. DIVERCITY had what the industry wants in real construction life.

6. Discussion on Lessons Learnt

Some of the main key outputs and lessons learnt from the DIVERCITY tests explained below, which can be beneficial for testing the innovative systems.

1. The DIVERCITY tests were not only done for verification and validation of the system but they were also done to introduce it to the end users who are not well familiar with such an innovation of computer-integrated construction.
2. Secondly, a process of workflow called storyboard was derived from the user requirements, which implies construction supply chain for lean construction work environment. All tests were conducted according to the storyboard by the end users not the developers. Based on the consecutive test results, the developers released subsequent prototypes until the end users were satisfied in the defined test criteria in every phase.

The use of storyboard was of significant importance in a problem domain where the scope is very broad and ill understood. In the case of DIVERCITY, the applications are pushing the boundaries of current construction industry practices. The industry was not clear on the requirements and applicability of the novel products, during the early stages of the research. The storyboard and the subsequent more detailed scenarios played a major part in requirements capture as well as the testing of the DIVERCITY.

3. The tests were undertaken in different countries at the same period of times in a distributed manner, which brought an extra novelty to the testing approach but also brought about extra difficulty in managing the test

activities. Because the test teams were geographically distributed, the test plans were communicated and a consensus should be reached prior to and during every testing phase. Under this circumstance, communication and interaction between the testers distributed across Europe were extremely important for effective tests. Another dimension of the difficulty of distributed testing was that test teams in different countries conducted the tests independently that resulted in different interpretations in discrete countries. These various interpretations were required to be harmonised and a shared interpretation of test results needed to be provided to the developers.

4. Business strategy for marketing and exploitation were executed more effectively due to the DIVERCITY tests. To be precise, business strategy and exploitation were not clear enough before the tests started. The DIVERCITY tests enabled the project team to clearly define the business strategy for marketing and exploitation of the system.

Furthermore, this shared understanding evolved over time; therefore the test results became increasingly more coherent.

As a result of the testing approach, the followings, which were already proved by researchers and engineers, were confirmed by means of the DIVERCITY tests.

- Initially, testing started early in the development lifecycle, and
- Tests were conducted iteratively and test teams continuously interacted with each other and the developers. These led us to close the quality gap in the focused testing objectives.
- Possible risks were identified and reduced earlier. Repairs to problems were less costly than repairs at the closing stages. Project was completed on time as desired.
- There was an iterative process between the requirements capture and software testing. Many of the detailed requirements, which were ill understood during the requirements capture, became clearer, and expanded during the testing.

7. Summary

In this paper, the case study for a system testing applied to the DIVERCITY system is explained. However, DIVERCITY consortium viewed the tests as a broad workflow incorporating continuous processes of tests focused on identifying and eliminating defects and observing the enhancements, assessing products quality in many respects such as functionality, usability, integration, reliability performance etc.

The user requirements were the key tools for DIVERCITY tests. While the requirements were designed as wish list of the end users at the alpha phase, at the beta phase they were addressed in the storyboard that is related to construction supply chain. In conjunction with addressing the user requirements, it also requires testing the integration and distribution issues. At the final phase, the storyboard was again the key tool. However, moving

from the beta to final phase, the test scope was broadened as it had happened moving from the alpha phase to the beta phase. Hence, the storyboard used in the final phase entails testing the advanced form of integration that refers to synchronise and asynchronies collaboration. The end users reactions to the system are evaluated.

Some key outputs and benefits were also highlighted such as the construction end users were being familiarised with the innovative systems, a process for testing was defined, which was an applicable process to the construction industry when innovative integrated information systems were employed by the industrialist, managing and conducting the tests by the end users in a distributed manner for large scale integrated information system and business strategy and exploitation plan of the system were clarified through the DIVERCITY tests.

References

Kindrick, J., Sauter, J., Matthews, R., "Improving Conformance and Interoperability Testing," StandardView, May 1996

Whittaker, J.A., What is software Testing? And Why is it so hard?, Florida Institute of Technology, IEEE Software January/February 2000

The rational Approach to Automated Testing, A Rational Software white paper, 1999, USA, *www.rational.com/products/whitepapers/100581.jsp*

Kruchten, P., The Rational Unified Process-An introduction, Addison Wesley, Longman, 1999.

William E. Lewis, Software Testing and Continuous Quality improvement, Software Testing Associates, Inc. Plano Texas, 2000

Alshawi, M. "*SPACE: integrated environment*" Internal Paper, University of Salford, July 1996.

Morris, KC, Flatter, D., Standards Based Software Testing in a Net-Centric World, Proceedings of the Software Technology and Engineering Practice, Pittsburgh, Pennsylvania 30 August - 2 September, 1999

Aouad, G., Marir F., Child T., Brandon P. and Kawooya A. "*Construction Integrated Databases- Linking design, planning and estimating, The OSCON approach*"; International Conference on the Rehabilitation and Development of Civil Engineering Infrastructures. American University of Beirut, pp 52-60, June 1997.

Issa, R., (1999) "*Virtual Reality: A Solution to Seamless Technology Integration in the AEC Industry*", Berkeley-Stanford CE&M Workshop, Stanford 1999.

Faraj I. and Alshawi M. (1999), A Modularised Integrated Computer Environment For the Construction Industry: SPACE, 1999, University of Salford, UK, http://www.itcon.org/1999/3/