

SDLC CHALLENGES IN DEVELOPING COUNTRIES NICARAGUA AND HAITI – A COMPARATIVE STUDY

Winston Tellis, Fairfield University, winston@fair1.fairfield.edu

ABSTRACT

Lee and Tellis (1) presented a modified SDLC for developing countries based on the work of Whitten and Bentley (2). This study extends that research to include Nicaragua. The paper compares the infrastructure between Haiti and Nicaragua. The report confirms the earlier findings of Lee and Tellis(1), but identifies the differences in each country and how they might affect the SDLC.

Keywords: SDLC, SDLC Developing countries, Haiti, Nicaragua

INTRODUCTION

Lee and Tellis (1) reported on the application of the FAST methodology developed by Whitten and Bentley (2). That application was in Haiti a developing country, and Lee and Tellis recommended several modifications to the FAST methodology. Resources often assumed in developed countries, are not available in developing countries. In Haiti electrical power was unreliable, transportation was slow and erratic, and even telephones were not freely available, and also not reliable. All those factors had an impact on the System Development Life Cycle (SDLC) and the design of the project.

Lee and Tellis also reported on the problem of language interface in the computer software. In Haiti, Kreyole is widely spoken, but is a phonetic language. No software has been developed with a Kreyole interface, hence the potential for acquiring such a package for any user community, is remote. The two authors researched and recommended software that had an English language interface, but had a numeric menu, which allowed the local users to interpret the commands correctly.

This study extends the Lee and Tellis(1) work, by applying their modified methodology in a different country, with a different cultural history. In applying the modified methodology in Nicaragua, the author found significant differences in several aspects of the procedure. Spanish is the language of commerce in Nicaragua, making it easier to locate potential software vendors. Although Nicaragua is considered the poorest country in Central America, it is substantially more developed than Haiti. The infrastructure of Nicaragua permits a level of commerce not unlike some of the smaller rural communities in western countries. "Nicaragua is not a poor country" was a comment by Mr. Harold Chavarria, Director of The Sister Cities Project in Leon in describing the resources available in Nicaragua.

The organization in Nicaragua is Centro de Promocion del Desarrollo Local (Ceprodel), and is involved in microlending to the organized poor. Ceprodel also lends to individuals with collateral, mainly farmers, and some organizations. Ceprodel was founded in 1991 with foreign loans and grants to foster local development, technology, and social programs. With persistently high unemployment, and reduced government involvement, the main problem is obtaining food for the poor. Their programs seek to create jobs, and income for their members. They have

recently made loans totaling \$5,000 in \$100 amounts. Earlier funds were used to make loans for housing, small business, electricity, water, and sewage.

Ceprodel needed an automated system to do the interest calculation for their microlending accounts ("portfolios" as they are called locally), currently being done by hand. An employee was assigned the task, with specifications from the director. The programmer developed the database in FoxPro, and has not had the time or training to convert to another more robust database product. The organization has not had a software upgrade recently since the employee is busy, and they do not have a backup person, if the programmer's employment was terminated. According to the former director of Ceprodel, the acquisition of accounting software exhausted their funds, but they still needed a portfolio management program. They did not follow any formal SDLC, rather they knew what they needed and provided specification to the programmer as the system was being developed. Since the employee was already on the payroll, no further funds were needed for system development. They did not consider the cost of the employee's time for development of the system as a cost.

In Haiti, Fonkoze found it necessary to expand beyond microcredit into fostering small businesses in rural communities. Similarly Ceprodel as described above branched out into economic development to assist members with loans that were unavailable from conventional banks. In both cases, the quality of housing of the members improved, while rural members started small businesses.

All the offices of Ceprodel have electricity, unlike the branch offices in Haiti, and more importantly, all of them also have a modem, which is used to dial the server at the main office in Managua. This is an important element in the ability to keep the central office database updated daily, which is vital to the overall management of the credit operation. Although there was a small LAN in operation at the main office, only one PC was used to manage the microloan activity in Leon. Three employees based in the main office in Managua, who are paid \$200 per month, maintain the LAN and the microloan system for all the offices. In contrast, the branch managers in Haiti usually have to take a public bus to Port-au-Prince to deliver the week's transactions, a trip that can take over 4 hours each way. There are seven PC units in Ceprodel's Leon office all connected to a server in a LAN. Only one PC runs the microloan software. It has a modem that allows the user to dial the Managua office to update the main database. The server allows for future addition of PCs if the loan activity increases.

System development is a critical element in developing countries as it is in developed countries. It is interesting to note that "most, if not all of the empirical evidence on information systems ... is confined to data from developed countries" (3). This suggests that even in today's global business environment, system development might be based on factors that are not entirely appropriate in a nonwestern environment. This is confirmed by Jeffrey (4) in his work addressing the essential difficulties of software engineering. He asserts that a system is "built by one group of people for another group of people, who differ from them in at least one, usually very important way" which might include language, work habits, and culture. In acquiring software for use in a nonwestern cultural environment, many complex factors must be considered. No specific recommendations for developing countries emerge from the literature. Even the work of Yuval (5) in which a matrix approach to system development is explored, would have to be modified to accommodate the existing conditions in a developing country.

The author has viewed and recorded the significant differences between developed and developing countries in previous research. In this paper the comparison between the system development process in two developing countries presents some interesting differences. The

prevailing infrastructure within the host country is an important element in the development methodology, and one which is not considered by any of the published methodologies. The common life cycle explored by Alonso et al (6) shows some promise in terms of defining goals and constraints. As far as resource and infrastructure deficiencies can be defined, the method could be useful. However, if the conditions are not clearly definable, then the problem is considered "ill-structured." A detailed analysis of eight development approaches by Iivari and Hirschheim (7) provide some indication that older methodologies tend to be technical-mechanistic whereas the emerging methodologies are more socio-technical. Clearly, the consideration of the social aspects will help to globalize the development process itself, and make it more generally applicable. This would address the concern that system development in a global enterprise must be done locally (8). Such local development deals with data format standardization, software and hardware configurations and local constraints that could impact the project.

In this paper the author will review the application of the FAST methodology developed by Whitten and Bentley (2) as modified for use in Haiti, and extend its application in Nicaragua. Comparisons of the relative infrastructure will be made insofar as there is an impact on system development.

COMPARISON OF APPLICATION OF THE FAST METHOD IN HAITI AND NICARAGUA

With the shortage of technical and support personnel in Haiti, system development for organizations (especially for non-profit organizations) was in the hands of international volunteers. In addition to the problems suggested by Flynn (9), financial and long-term support and maintenance were almost non-existent. To make matters worse, end-users of SDLC projects could not directly communicate their needs and expectations to the systems development team. Lee and Tellis implemented the Whitten and Bentley (2) FAST methodology in Haiti with modifications. The Nicaragua implementation was different in that a system analysis was done by a large international organization. There was a staff of technical people in the main office in Managua available to all the offices. The language of commerce and technology was Spanish, and the software was developed with Spanish interfaces. The FAST methodology was applied as follows:

1. *Survey and Plan the Project*— The Director and one assistant described system needs. Fondasyon Kole Zepol (Fonkoze), acts as a credit union. There were over 600 accounts in Port-au-Prince in 1997. Most transactions were for under \$50.00. All communication was in Kreyole, a phonetic language. Ceprodel also had about 600 accounts, with most transactions under \$100.00. The World Bank designed the system for Ceprodel and required certain reports. Their design was not consistent with Ceprodel's business practice. Ceprodel assigned an employee to develop the system "in-house".
2. *Study and Analyze the Existing System*--The current system was manual in both cases and was susceptible to human errors. The new system, with a numeric menu (for Haiti), and in Spanish for Nicaragua, would create an error-free TPS for the organization. All interest calculations were made using a hand calculator.
3. *Define and Prioritize the Business Requirements* --In Haiti, the project was tested with the activity from the Head Office only. Subsequently, the accounts from the branch offices were

- added. It also calculated the interest, payment amounts and schedule. In addition, it provided a printed report upon request. A single PC in the main office handled the transactions. The system will eventually be networked with the branch offices to create a master file of loan transactions. In Nicaragua, The World Bank defined the requirements, which were modified to reflect Business practices at Ceprodel. They needed urgent assistance in loan management.
4. *Configure a Feasible Business Solution*--Frequent power outage in Haiti means that a back up power supply is needed to secure the operations of the system. Serious hardware problems would require the equipment to be sent to Miami, Florida for repair. In Nicaragua, there was more access to technical help, and employees were trained in support and network maintenance.
 5. *Procure any New Hardware or Software*—Fonkoze had a budget of \$2,500 to acquire hardware and software. Four vendors presented bids for Credit Union management systems. The successful bidder also confirmed the choice of hardware platform. None of the hardware and software vendors would provide local service in Haiti. Ceprodel decided to develop the system in-house without a formal methodology. They had no budget, and so used an employee's services to accomplish the task of system development.
 6. *Design and Integrate the Target System*--Since there was no existing computer system, integration was relatively simple. Designing the user interface, though, was a difficult task for Fonkoze. Since there was no local user support, and error messages in English are unreadable by the users, a numeric menu was imperative. Users were trained to halt work if any message appeared on the screen. Ceprodel implemented modules as they became available. The database was developed first, and then the access programs.
 7. *Construct and Test the Target System*--The team in Haiti decided to implement the new system in phases. At Ceprodel, they also implemented what was ready, since they had no prepared plan.
 8. *Install and Deliver the Production System*—In Haiti, the system was installed and tested by a volunteer using vendor supplied procedures. The system has been operating for two years without any problems. The U.S. vendor provides Fonkoze with periodic upgrades. In Nicaragua, there is no sense of completion, because no formal development methodology was used.

CONCLUSIONS

The Haiti study showed that an SDLC designed for a developed country needs modification before it is usable in a developing country. Thus the Whitten and Bentley SDLC was modified by Lee and Tellis for a successful implementation in Haiti. In extending that study the author has shown that even a methodology developed by an experienced organization such as The World Bank, was not suitable for the particular organization in Nicaragua. The nature of microlending was the same in the two countries, but the state of the infrastructure could not be more different. The methodology used in Haiti was successful, and continues to function normally, with frequent calls to the United States for vendor assistance. The Nicaraguan project took a different path to SDLC, and also had a successful implementation.

It is clear that although both are developing countries, the resources and infrastructure available in Nicaragua are greater. Trained personnel make the Nicaraguan project more easily manageable. However, Ceprodel's decision to develop the software internally, and not have a long-term support plan could be a serious problem in the future. While the Haiti project receives

regular software updates and assistance from the vendor, the Nicaragua project has had no upgrades since its implementation in 1997.

In the future system development is likely to increase in importance in developing countries where the majority of the world's population resides. Major software corporations currently subcontract some of the development work to developing countries such as India. Hence the importance of research into SDLC will likewise increase, and become more attuned to the realities of the developing countries.

Table 1

	Haiti Project	Nicaragua Project
<i>Survey and Plan the Project</i>	Project includes Director, technical staff member, and user. (Director speaks English while the others speak Kreyole--a phonetic language.)	Director, programmer. Spanish interface available. Hardware in place.
<i>Study and Analyze the Existing System</i>	Current system is recorded manually and in Quicken. 600 accounts. End-users provide minimal input because of language problems. New system to automate the recording of loan transactions. Reporting required.	Manual system. 600 accounts. Recording to be replaced by database system. World Bank needs reports.
<i>Define and Prioritize the Business Requirements</i>	The new system will produce files for payment deposits, and other cash transactions. Easy to use with minimal instructions and supervision. Numeric interface.	Director specified design to programmer. Interest calculations to be automated. Spanish interface acceptable.
<i>Configure a Feasible Business Solution</i>	Quicken has no double entry, inadequate field lengths. New system requires an accounting setup. Frequent power failures can interrupt operations.	Infrastructure in good shape. Programmer needs clear design instructions. New system must handle portfolio management. Reporting required by World Bank, others.
<i>Procure any New Hardware or Software</i>	Acquire PC to load software. Backup power supply to provide uninterrupted data entry and reports. New system will use database technology.	No new hardware required. Software to be developed by staff member.
<i>Design and Integrate the Target System</i>	New system will create reports identified in Design phase. Reports only be available to Director, since system produces English reports.	System to be delivered as each module is ready. Database to be designed and access programs as needed.
<i>Construct and Test the Target System</i>	Software installed, tested per vendor materials. User data loaded, tested. Arrangements made to call USA for vendor support.	The system was delivered in modules. The main office was used as test site. If successful then other sites will receive module.
<i>Install and Deliver the Production System</i>	Installation of system was simple with existing software. Training of key personnel successful. Local language interface complicates training of staff.	Spanish interface expedites training. All users are literate.

REFERENCES

1. Lee, P, Tellis, W. (1998). Systems Development Life Cycle in a Developing Country: A Constrained Model. IACIS Conference 1998.
2. Whitten, J. L., and Bentley, L. D. (1998). Systems Analysis and Design Methods, 4thEd., MA:McGraw-Hill/Irwin.
3. Khalil, O. and Elkordy, M. (1996). Relationship between Some Decision and Task Environment Characteristics and Information Systems Effectiveness: Empirical Evidence From Egypt. ACM/SIGMIS '96, Denver, CO.
4. Jeffrey, H. J. (1996). Addressing the Essential Differences in Software Engineering. Journal of Systems and Software, (32), p. 157-179.
5. Yuval, A. (1996). The Matrix Approach to Information System Development. Computers in Industry, (28), p. 257-263.
6. Alonso, F., Juristo, N., Mate, J., Pazos, J. (1996). Software Engineering and Knowledge Engineering: Towards a Common Life Cycle. Journal of Systems and Software, (33), p. 65-179.
7. Iivari, J, Hirschheim, R. (1996). Analyzing Information Systems Development: A Comparison and Analysis of Eight IS Development Approaches. Information Systems, 21(7) p. 551-575.
8. Barker, R. (1993). Information System Development in a Global Environment. Business Forum, 18, (1), p. 57-59.
9. Flynn, J. A., (1997, December). Global Information Systems: Problems, Solutions, and How to Manage Them, JISE, 6 (3), 3.