# A Case Study of Successful Geographically Separated Teamwork

Dewayne E. Perry, Gurgit S. Gil, and Lawrence G. Votta
Bell Laboratories, Lucent Technologies

## *1. Introduction*

Increasingly, software development organizations are separated by geographical, temporal and cultural boundaries. This puts stress on the software development processes and organizations because there are fewer direct interactions and greater possibilities for misunderstandings.

In this case study, we look at one particular team that was separated by country boundaries, a one hour time difference and nationality cultural differences. Despite these differences, the test team was considered to be ''successful'' where other teams in the same organization spread out over the two locations were not so considered.

The purpose of this case study is to isolate the factors for that success and provide a path for successful team cooperation across geographical, temporal and cultural boundaries.

We first give an overview of the project, its organizational and geographical structure. Before proceeding to the details of the case study, we discuss our approach and how we gathered the data for the case study. We then discuss the study teams' process and its demographics. At the heart of the study are the basic factors — that is, what is meant by ''successful'' — and the various contributing factors — that is, what are the factors that were critical in producing the ''success'' of the team. We then summarize the lessons learned from this case study for creating successful geographically separated teams and discuss some of the as yet unanswered questions raised in the context of this study.

## *2. Overview of the Study Project*

The context of our study is a new product development in a new organization and based on a legacy product. The product itself is a real-time, embedded communications product integrating both hardware and software components into a functioning system that interfaces between people other Lucent Technologies products.

The development under investigation is an initial system release of the software that forms the heart of the combined software and hardware product. The development teams are located in two different countries, where one team is made up primarily of new people brought in specifically for this project (often with application domain knowledge gained through similar or related development projects), and the other team is made up of people with experience in the legacy product on which this initial release was based.

Characterizing the project, then, are the problems of geographical, temporal and cultural separation. The two sites are separated temporally by one hour and geographically by about 700 miles. Cultural differences range from different languages, though the official language of the project is English, to different work cultures and governmental work regulations.

The project suffered the usual start-up problems: a new product with its sets of unknown problems, new people that are not used to working together, no well-understood and defined processes, and new management with its as yet unknown management styles. In short, a typical ''time to market'' pressures to build a product (albeit based on an existing legacy product) with an untried organization whose

development is complicated by geographical, temporal and cultural separation.

## *3. Our Approach*

As this is primarily a retrospective study, and given the nature of the of the questions to be answered, it is of necessity a qualitative, rather than quantitative, study [1, 2]. None the less, we have assembled a convincing case of support for what, in this case at least, were the critical factors in success of the test team as a well-functioning multi-site team.

As the basis for our study we collected data from various sources:

- the project retrospective,
- the test team retrospective,
- project documents, and
- semi-structured interviews.

We were fortunate in having both project and test team retrospectives. It is from the results of these retrospective exercises that the question arose as to what made the test team a successful collaborating team. In particular, these documents provided us with a rich history of the project, both in terms of what was considered to have been done well and what was considered to have been done poorly. Moreover, they gave us a deep understanding of the project as a whole and where the test team fit into the development team and how they related to other parts of the project.

Project documents, of course, are among the basic products of the project. These give us the domain of the project and product and are, in part, the basis for the test team's test strategies and plans.

Our primary source of data, however, is our semi-structured interviews of testers, test team leaders, technical managers, and the overall project manger. We used prepared questionnaires to guide our interviews with the technical people and a set of general questions for the managers.

The questionnaire for the technical interviews (see Figure 1) focused primarily on gaining an understanding of the people involved, what they did and how they did it, what they considered to be successful or not successful, and how things evolved over the life of the project. This questionnaire proved to be extremely helpful in eliciting an understanding of what the test team did and how they did it. In particular, it was helpful in isolating the critical factors that contributed to their being a ''successfully collaborating team''.

At the heart of our approach is the following method which we used to establish our results which are described in the next sections.

- Determine what they did as their basic work
- Isolate and validate what ''successful collaboration'' means
- Determine the factors that contribute to that success
- Look at the literature for supporting models or theories

## *4. The Test Process*

We considered it to be of primary importance to understand what the test team did in order to determine what made it successful. As a result of our interviews, we formally defined the general test process that they used. We provide the visualizations of these processes here [3]. These processes then provide the basis for our analysis of their collaborating success and the factors that contribute to that success.

Dear Participant,

Thank you for agreeing to participate in our study of the . . . Release 1 Test Team. We would like you to think about the questions below in preparation for our interview with you in the next few days. We ask that you do not discuss these questions with any of your colleagues. All responses will be held in the strictist confidence. Results will only be available as statistical summaries.

1. Describe your professional experience and describe how you got involved with testing?
2. Let us separate the testing process and its results for the moment.
   - What are the main testing processes?
   - What are your roles and responsibilities?
   - What are the activities that you do? When do you do them? What are your entrance criteria for these activities?
   - What other software development processes do you interface with?
   - How do you create your tests? What are your sources?
   - What tools and technology do you use?
   - What is your strategy for successful testing?
   - What support do you need to execute your tests (eg, test lab, system expertize, etc)?
   - What are your exit criteria (ie, how do you know when you are done)?
3. Do you consider testing to be successful? The testing process? What are the contributing factors?
   - People, Experience, Education and training, The process, Technology
   - Project plans and schedules, Resource allocation, Management, Management style
   - Social factors, Cultural factors, Organizational Factors
4. What factors have hindered the success of testing?
5. What factors have evolved over time? For the better? For the worse?
6. What do you see currently as critical problems?

**Figure 1**: The technical interview questionnaire

---

The testing of this kind of communication product requires a complex physical test laboratory, with both simulated and actual hardware support. Not surprisingly configuring the test lab is an important part of the testing process. Establishing test laboratories at both sites was of critical importance to the entire process of testing in its various forms.

We divide the visualizations of the test process into two segments: a generic test plan creation process and the full set of testing processes. The generic test creation process defines a basic process followed for creating the different bring-up, specification, performance, adversarial, regression and stability test plans. The inputs to the test creation processes come from various sources: requirements, design documents, product integration entrance requirements, and the customer documentation. In the visualization in Figure 2, ‘xx’ stands for one of the types of test plans. The syntax of the formal descriptions is simple: each process is defined by a name, a set of inputs (artifact and source) and outputs (artifact and destination). The visualization is derived from the formal description.

Of the sources for the test plans and test designs, the customer documentation, the product integration entrance requirements, and the meeting with the product integration personnel were the most useful. The requirements and system design documents were the least useful. While this may seem the reverse of what ought to be the normal case, it is not unusual in a start-up project for the requirements and system design documents to be in a state of flux during the early parts of the project when the test team was creating their

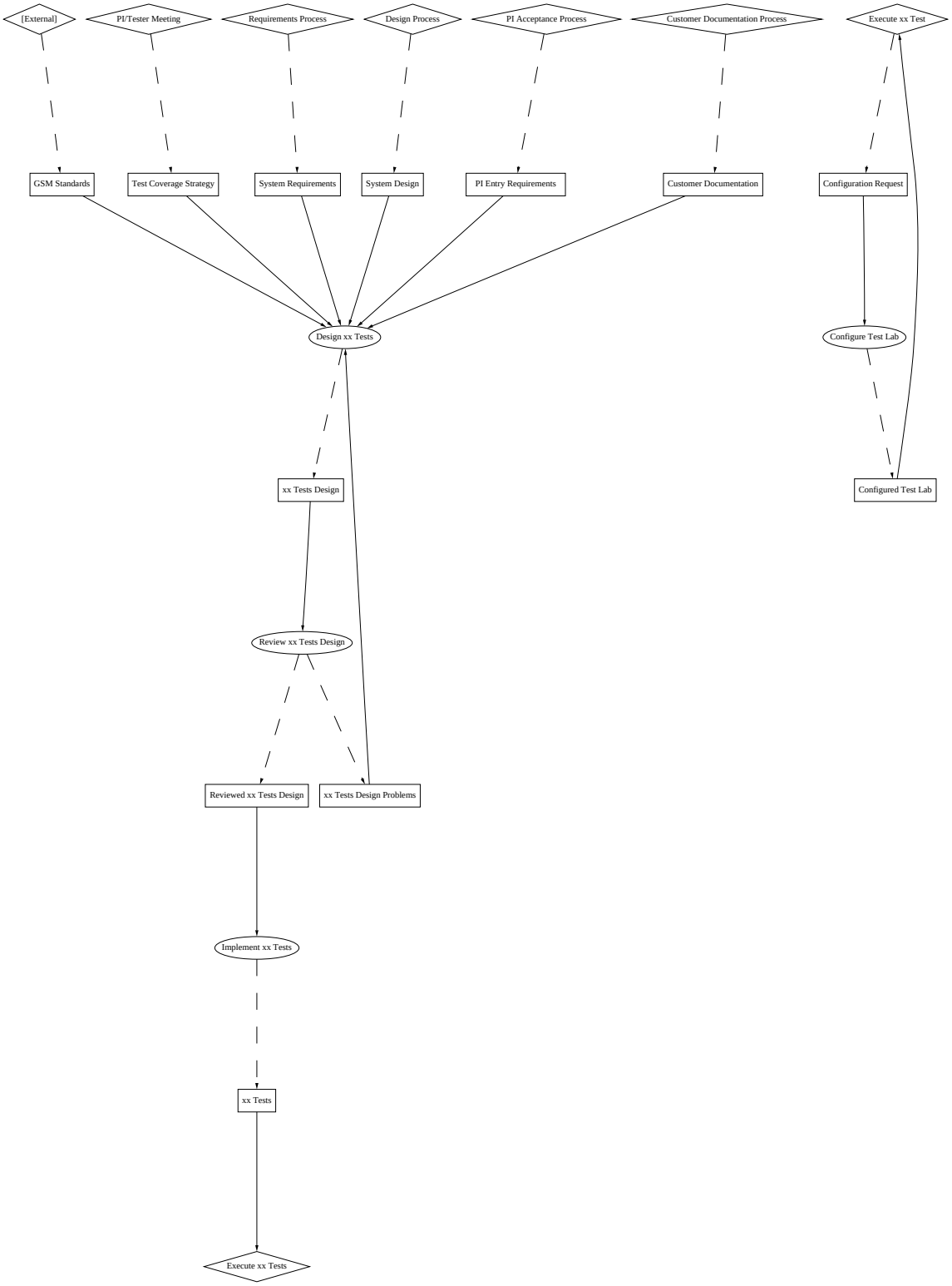


**Figure 2:** The generic test creation process and the lab configuration process. The external processes are depicted in diamonds, the modeled processes in ellipses, and the input and output artifacts as rectangles. Output relations are dashed arrows, input relations are solid arrows.

test plans.

The test processes are pretty much independent of each other with two exceptions. All of the test processes are dependent on the sanity-test process, and the stability test process is dependent on the regression test process. There are obviously other interdependencies between the test processes, but they are informal ones that are not explicitly defined.

The sanity, specification, performance and adversarial were performed at one sight, the regression and stability tests an the other site.

In terms of test support, the tests were in the beginning run manually. Over time, test automation was introduced as part of the test process and an increasing amount of the tests were run automatically. This was of particular importance for regression testing.

### *5. Successful Collaboration*

‘‘Being Successful’’ in the context of our study can be interpreted in several ways. First, it can be interpreted as doing a good job — that is, that the team did the testing well and produced a high quality product. While this may well have been the case, it is not the focus of what the managers had in mind with their judgement. The meaning they had in mind instead was that they were a ‘‘good working team’’ despite the problems of being located in two different countries with all that that implied.

The important question then is ‘‘What is an operational definition of begin successful in the sense of being a ‘good working team’? ’’ Towards this end, we isolated and confirmed three critical factors that define what was meant by the testing team ‘‘being successful’’.

- They got the work done.
- They did effective planning.
- They were self-managing.

These factors were derived primarily from interviewing the team leaders and technical managers. At the end of our set of interviews we talked with the project manager (using semi-structured interviewing approach in much the same way we had for the rest of the test team related interviews) to find out what his view of what were the critical factors determining the test team’s success. His response either matched our factors exactly or were congruent with them. Thus we consider the success factors we isolated to have been validated by the critical source, the project manager.

### *6. Contributing Factors*

The next important question then is ‘‘How did the team achieve these three criteria for successfully working together?’’ What were the critical factors that made their achievement possible and how did they exploit these factors?

We classify the underlying mechanisms that comprise the significant contributing factors in this study into three classes:

- Managerial strategy
- Team building process
- Effective process execution

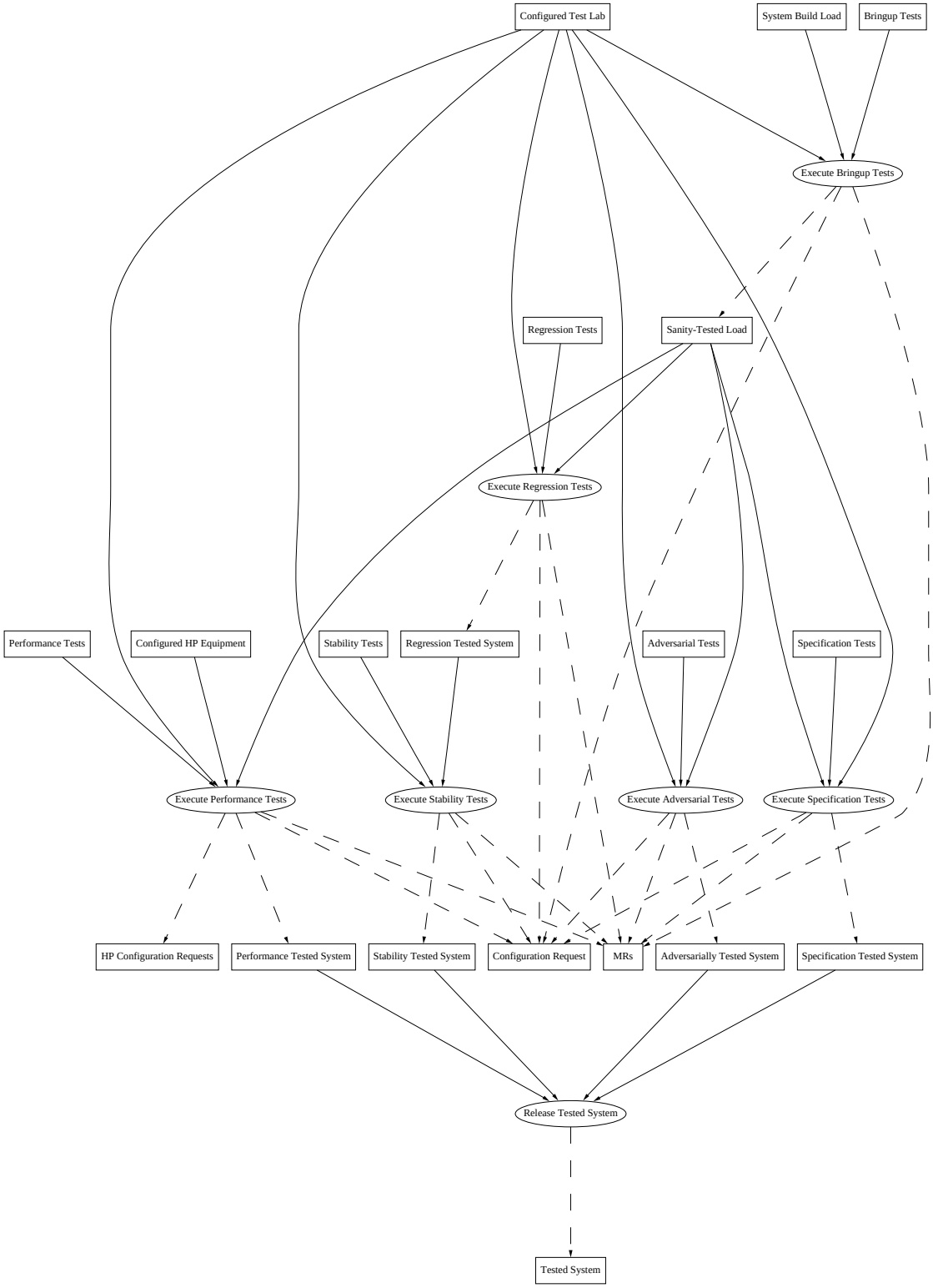


**Figure 3:** The full test process with the individual testing sub-processes. To simplify the visualization, the external sources (the test plan creation processes) and destinations (the configure lab and product integration processes) have been eliminated.

### *6.1 Managerial Strategy*

Not surprisingly, managerial strategy factors play a significant part in the determining factors for the successful working of the team. The initial factor in the process of successfully working together is the managerial creation of a context for self-management. In other words, the test team leaders were provided a context in which they could derive there own approach to getting their work done. Whether this management strategy was the result of keen insight into successful team formation or a result of having too much to do (a common enough situation) is immaterial. The stage was set for the team leaders to manage their own affairs.

Within that context, the team leaders created a shared vision of what needed to be done. They agreed on what the basic goals of the team were, what the critical needs for the team were, and how to go about satisfying those goals and needs.

Having established the basic plans for the test team, they then went about actively working critical resource issues (in this case, the test laboratory and the sources of their test plans). Rather than wait to react when crises arose, the team leaders were proactive in working those resource issues before they reached a stage where they were in crisis.

Furthermore, as a part of the management strategy of working those critical resource issues, they divided them up between the two sub-teams in order to optimize both the resource issue solutions, but also to optimize the work via local (team leader) management.

Thus, the test team leaders fruitfully exploited the context for self-management by successfully managing themselves.

### *6.2 Team Building Process*

In the paper ‘‘Time Matters in Groups’’ [4], the authors define a team formation model [5] that delineates a sequence of stages of group activity (goal choice, means choice, policy choice, and goal attainment). The underlying historical sequence of events for the testing team closely follows this model. It is our hypothesis that the testing team had sufficient time to build up trust and follow all the stages in this team formation model, whereas other teams, often because of critical path considerations, was not able to do so.

Certainly one critical factor in their process of successful team formation was that they were not on the critical path in the early development of the product. There was time to work out various impediments and disagreements. By the time they were on the critical path for testing the components developed at the two sites, they had successfully set the stage to be able to do their job.

Historical accident or not, the test team had time to build the trust in each other that is necessary for successful team formation. The primary means for doing this was the creation of the test laboratory. As we have mentioned in the preceding section, the team leaders had a shared vision for what needed to be done as part of their job. The creation of the test lab was their primary focal point and shared goal. As both team leaders had had experience with similar products and their testing requirements, it was clear to both of them that the test lab was the most critical resource they needed to successfully do their job. Interestingly, the lab technicians functioned as facilitators between the two subteams in the process of creating the test lab.

The initial interactions of the team were as a single unit working on a common goal. Once the team had reached the stage of having agreed on the goals and begun creating their critical resource, they then split into two independent, but mutually supportive teams that interacted well across the geographical, temporal and cultural boundaries.

Thus, there was time to build that necessary ingredient for successful team formation, namely trust.

### *6.3 Effective Process Management*

As we have noted in the preceding two sections, the test team effectively used its time off the critical path to ensure that their test processes would be as effective as possible by first establishing the test laboratory.

They also effectively worked other critical resource issues such as the sources for their test plans. When the usual sources proved to be unstable because of schedule and market pressures, they sought other sources for input to creating the test plans. They sought alternative solutions to critical problems instead of complaining about not being able to do their job.

Finally, the took a proactive role in improving their process and test support: the moved from manual testing to automated testing. Since testing is a labor intensive process, automation can in many cases provide significant process improvements. And of course, in the case of regression testing much of the work is repetitive, automation can be an especially attractive source of making the testers more effective. Two benefits were accrued: their manual work was made more effective, their testing effectiveness more uniform, and the testing cycle reduced by a significant factor.

Thus, not only did the manage themselves, not only did they plan well for their work, not only did they get their work done, but over time they did their work better and faster.

## *7. Lessons Learned*

There following are the primary lessons learned from this study:

- Time is necessary to build trust
- Face to Face contact is necessary to build working relationships
- A clear agreement on team goals is necessary to coordinate the geographically separated teams
- It is critical to match communications to the intersite interfaces.

One of the primary facts about the our study development is that the test team had enough time off the critical path to effectively follow the team formation model and build trust among the critical members of the team.

While it is possible with electronic means to establish working relationships, it was a consistent comment throughout our interviews that face to face contact was a precondition to building effective working relationships. Having a face to attach to the other side of a communication process personalizes it and gives it an immediacy that it would otherwise not have.

While the agreement on the team goals may well be an essential ingredient in building trust amongst the team members, it is also critical for the execution of the geographically separated teamwork. Without that essential agreement, the sub-teams cannot work effectively as independent entities.

The final essential ingredient in effective multi-site teamwork is the necessary bandwidth to support the interfaces among the subteams. Otherwise, too much time is lost in non-productive workarounds that both slow the team down, but also erode the working relationships.

We conclude that geographically distributed teams can work as long as they have time to follow the team formation model and as long as there is appropriate engineering of inter-site interfaces and communication.

## *8. Validity*

We argue for internal validity in the following ways. First, we used as many non-intrusive means of gaining data as possible (project documentation, project and test retrospectives, etc). For further data gathering, we used the well understood qualitative technique of semi-structured interviews.

Second, we covered a very good cross section of the test team in our semi-structured interviews, interviewing testers, test team leaders, technical managers as well as the manager of the entire project. Thus we have the entire spectrum of views represented in the set of interviewees.

Third, during our interviews we tried very hard to account for ‘‘research effects’’ (see Grinter and Herbsleb [2] for various threats to internal validity in qualitative studies). We followed standard good practices in our interviews being careful not to lead the interviewees, but to elicit their responses in their own words and terms.

Finally, we first extracted what we considered to be the model of what was considered to be successful in this particular study from the interviews and confirmed that model with an interview with the project manager to elicit what he considered the success factors to be. Our model was both consistent and congruent with his model.

We argue for the external validity of our study, at least for the class of real-time embedded systems, in the following ways. First, for this class of systems, this is a pretty typical project: the mix of software and hardware development, the presence of an extensive system test laboratory, the pressure of market conditions, etc.

Second, this part of the development and evolution process is probably the most uniform across this class of systems. Where in requirements analysis, and system architecture and design, we find a great degree of variability in the methods and techniques used, in the testing part of the process there is little of the wide diversity.

Third, given the globality of both the market place and companies competing in that market place, the issues of multi-site development are of critical importance. However, even without the complicating factor of multi-site development, the lessons here are applicable to the basic problem of successful team formation.

Fourth, a major part of the factors that contributed to the successful working of the test team were such that they followed a known model of successful team formation [4]. This argues for the utility of the team formation process and for a more general interpretation of that part of the study beyond the restrictions of the more focused set of real-time embedded systems.

Fifth, since a start-up project presents almost the worst conditions under which to effectively form successful teams, we argue that our study has a broad applicability: if this study is valid in these restricted conditions, then they also will be valid under less restricted conditions. Of course, these restricted conditions were ameliorated for the test team since they were not on the critical path. However, we feel that that is in fact the critical factor and indeed that which differentiated the successful from the non-successful.

Finally, the early indications from another study of other Lucent Technologies multi-site development projects [6] are congruent with this study.

There are two factors for external validity that may somewhat limit the generality of our results. First, most projects are not in start-up mode, but in evolution mode. We argue that there are aspects of both modes present here: part of the project was a legacy project in which the product may be viewed as an evolutionary step; part was a new project in which the start was essentially from scratch, certainly in terms of people and

the project itself. However, at this point it is certainly true that the project is no longer in start-up mode.

Second, our notion of ‘‘success’’ was a very limited one, limited to the successful working of a multi-site team with little of the qualification of how well they did their job itself. However, we argue that it is domain specific factors that are the major part of this richer notion of ‘‘success’’ and that these are orthogonal to the ‘‘successful teamwork’’ factors which we focused on.

## *9. Open Questions*

These last issues of validity lead to a number of questions that were not answered by this study, primarily because they were beyond its scope. None the less, they are important questions and need to be considered in further studies.

A common occurrence in the on-going execution of a project is the addition of new people to the project. How do you incorporate someone new into an existing team? What is a useful team absorption process and is it different from the initial team formation process? If so, in what ways? Is it a faster process or a longer one?

Another common occurrence is project reorganization. How do you reorganize and retain successful teams? Do you have to go through the entire team formation process, or is there a different team reformation process? Is it faster that the initial process?

And of course, probably the most important question of all: how do you rebuild teams where they have already failed? How do you transform into successful teams those teams that did not have time to successfully form?

## *Acknowledgements*

We thank especially Frank Wales for his depth of knowledge about the project, his valuable suggestions and insights into the important factors in this study, and his unflagging support in arranging many of our interactions with the test team, technical and project management. We also thank the test team for their cooperation, input and feedback. Finally, we thank the rest of the Collaboratory Project team for their help and comments.

## *References*

[1] Barney G. Glaser and Anselm L. Strauss. *The Discovery of Grounded Theory: strategies for qualitative research*, New York: Aldine de Gruyter, 1967.

[2] Rebecca E. Grinter and James D. Herbsleb. ‘‘Qualitative Methods in Software Engineering’’, 1998. Submitted for publication.

[3] David Carr and Ashok Dandekar and Dewayne E. Perry, ‘‘Experiments in Process Interface Descriptions, Visualizations and Analyses’’, Software Process Technology, Fourth European Workshop -- EWSPT’95, Springer-Verlag, April 1995.

[4] Joseph E, McGrath. ‘‘Time Matters in Groups’’, in [5], pp 23-61.

[5] Jolene Galegher, Robert E. Kraut and Carmen Edigo. *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*, Hillsdale NJ: Lawrence Erlbaum Associates, Publishers, 1990.

[6] Rebecca E. Grinter, James D. Herbsleb and Dewayne E. Perry. ‘‘Lessons in Multi-Site Software Development’’. In preparation.