1 Introduction

This part of the paper discusses the cost estimation activities in CSELT and the up to date results of the collaborative project from the CSELT point of view. CSELT (Centro Studi E Laboratori Telecomunicazioni) is the research center of the Italian Telecommunications Holding Company (STET). One of the important current CSELT activities concern modern software technologies employed in the telecommunications network planning and management. In this contest, since 1990, CSELT has launched a project for the estimation of software costs.

The CSELT approach for the introduction of cost estimation in an organization that produces or purchases software consists of three major steps: (a) Assessment of the results obtained by the software supplier/buyer through the use of commercial tools or public models (such as COCOMO) and introduction of new tools and methods. (b) Extensive application of the new methods and systematic collection of data. As soon as possible, calibration of the commercial tools used on past data and generation of local cost models. (c) Application of calibrated cost models/tools and assessment of the overall approach.

Initially, the paper presents the CSELT approach to the problem which is based on an estimation process that has been defined and the use of a couple of commercial tools. A number of techniques for historical data analysis have been also defined and are currently under study including various algorithms for neural network training with historical project data, a case based reasoning system for estimation by analogy, factor and cluster analysis and an object-based technique for the generation of estimates in the
early phases of the life-cycle.

In parallel an integrated environment is under construction aiming to support data collection and estimate generation by merging together gathered historical data, commercial tools and analysis techniques.

The paper concludes by presenting the results of the application of the CSELT estimation process and commercial tools on the four projects of the Bellcore-CSELT collaboration. The experiment proved to be quite interesting and some lessons have been learned concerning the calibration of tools on a specific software production environment, sensitivity analysis and assessment of input data quality.

2 Cost estimation activities at CSELT

CSELT estimates mainly the cost of software products that are purchased by various departments of Telecom Italia (the national Italian telecommunications operator). In most cases the estimates are generated after the completion of the software product and are used in order to assess the project's productivity. When a sufficient number of projects is completed the overall productivity of the supplier can be also assessed. The cost estimation team has examined about 100 Italian telecommunications projects up to now. Data related to the costs of the above projects are systematically gathered in order to be used for future local cost model generation.

Commercial tools

The major estimating tool used in CSELT is the ELYS® system (although some other commercial tools are also under study). It is a product calibrated on french projects and it has been considered the most appropriate for the analysis of european projects. There exist two versions of ELYS: the real time version (ELYS-RT) and the business version (ELYS-B) [ELY] which is most suitable for data intensive applications (such as data base interactive programs). The former accepts project size in terms of number of lines of source code while the latter estimates project size based on the number of the elements of the conceptual data model, such as number of entities, attributes, relationships, as well as the number of screens, reports, interfaces etc.

Once the size of the project is determined a number of cost drivers is used to adjust the project effort. These cost drivers include COCOMO like cost drivers such as personnel experience, product characteristics and so on.
Special attention is given to the design methodology applied to the project, as well as to the requirements volatility and quality. Finally, a minimum effort and optimal schedule are computed. These results can be adjusted according to a more realistic schedule imposed by the user.

**Cost Estimation Process**

The cost estimation process for a project is shown in the following picture.

![Cost Estimation Process Diagram](image)

*Fig. 1. Estimate process and flow chart*
During the initial phase some important data are gathered in order to choose the method that will be applied and to plan the estimation activities. These data are:

- Brief description of the project history
- Project nature (e.g. real time, data intensive software or combination of more types of software)
- Project status (e.g. life cycle phase)
- Brief description of the functions that the software must accomplish
- Type of language and rough software size (e.g. in hundreds of Kilo lines of code or hundreds of Function Points)
- Various project management characteristics. Most important information is the number and kind of sub-projects to which the project is probably split up.
- Software supplier tools that can help calculate the size of the project (e.g. code counters that are available by the software supplier and the rules that are implemented through them, tools that can extract the number of Function Points from the source code or the number of elements of the logical data model)
- Supplier personnel experience with cost estimation and sizing methods and tools

Based on the above information the following choices are made:

(a) methodology for software sizing and estimation
(b) resource allocation (number of necessary estimators and estimation experience)
(c) planning of the estimation activities to follow

The data requested through the Questionnaires are the ratings (High, Average, ...) for the cost drivers that are necessary to run the commercial tool used, along with information on various other project data that can help the estimator produce a reliable estimate. The Questionnaires contain also some guidelines (that are enhanced with each new release) aiming to reduce the subjectivity needed to produce the answers to the questions asked.

The project people are first given some time (typically some days) to read and try to understand the kind of data that are requested through the Questionnaire. Normally they are able to fill the Questionnaire and sometimes produce a measuring of their software size. A meeting is then arranged aiming to resolve any problems arising from subtleties in some questions or data asked and review the answers provided. This method is
found to save a lot of time to the estimation process since all meeting parties know the topics to be discussed in advance.

After all data have been gathered the estimate generation takes place. An estimate report is produced containing, apart from the results, a description of the estimation activities, data that have been modified due to the interview, explanation for the various choices made and comments on the reliability of the estimate. The estimate is reviewed and is used to plan the project and/or for customer-supplier negotiations purposes.

Innovative methods

CSELT is also pursuing new methods for cost estimation and cost data analysis. Currently three research topics are under study and development.

(a) Cost data analysis through the use of neural networks. Neural networks are trained with historical cost data aiming to generate powerful cost models. The same technology is planned to be used for in order to reduce the number of input parameters cost drivers used in the model (dimensionality reduction) ([CAP93], [ROS94]).

(b) Analogy Based Cost Estimation. The characteristics of the project to be estimated are compared to the characteristics of projects in an historical data base trying to identify those projects with the closest similarity. The most similar projects are chosen according to an artificial intelligence technique known as Case Based Reasoning. An estimate is then produced with a simple mathematical formula that takes into account the size and complexity of the project being estimated ([BIS94]).

(c) Object based cost estimation. This is a cost estimation methodology aiming to produce reliable estimates on the basis of the user requirements specifications of software projects. The core of the method proposed is the object-oriented analysis of the specification documents in order to identify potential cost parameters among the elements that compose the resulting object-oriented model. The cost model to be used must have been previously generated through the statistical analysis of past project data. ([STA 94]).

The former two methods are validated on the COCOMO historical project data base and are are going to be applied on the CSELT historical data base. In addition CSELT is pursuing a methodology for Estimation Capability Assessment ([GLI93]) and Software Cost Model Quality Assessment.
Integrated Environment

An integrated environment is currently built at CSELT (fig. 2). The major components are the Cost Data Base, the Cost Data Analysis Tools, the Commercial Estimation Tools and the User Interface.

Fig. 2. Integrated environment

3 Estimation of the four projects by CSELT

The CSELT cost estimation process has been applied to the four collaboration projects. The estimates for projects A and B have been generated easily and the precision for the calibration releases was quite satisfactory as can be seen from tables 1 and 2. On the contrary some effort in cost modelling had to be spent on projects C and D due to some peculiarities that are explained in the relative paragraph.

Project A

Project A has been subdivided in 2 sub-projects respectively and the procedure presented before has been applied. No special problems arose in the estimation of the two releases using ELYS-RT. The results obtained are
shown in the following table.

<table>
<thead>
<tr>
<th>Life-cycle phase</th>
<th>Actual (staff Mths)</th>
<th>Nominal effort for each phase (%)</th>
<th>Estimate (staff Mths)</th>
<th>% Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reqs Anal.</td>
<td>3</td>
<td>5</td>
<td>3.57</td>
<td>+19</td>
</tr>
<tr>
<td>Design</td>
<td>19</td>
<td>28</td>
<td>20.3</td>
<td>-6.4</td>
</tr>
<tr>
<td>Code/Unit Test</td>
<td>30</td>
<td>42</td>
<td>30.3</td>
<td>+1</td>
</tr>
<tr>
<td>Integr./Syst. Test</td>
<td>18</td>
<td>25</td>
<td>17.98</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>70</td>
<td>100</td>
<td>72.15</td>
<td>+3</td>
</tr>
</tbody>
</table>

**Project B**

This is the smallest of the four projects. It is a single project (no subprojects) and has been estimated with ELYS-B. The results are shown in the following table.

<table>
<thead>
<tr>
<th>Life-cycle phase</th>
<th>Actual (staff Mths)</th>
<th>Nominal effort for each phase (%)</th>
<th>Estimate (staff Mths)</th>
<th>% Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reqs Anal.</td>
<td>1,4</td>
<td>7</td>
<td>1.34</td>
<td>-4</td>
</tr>
<tr>
<td>Design</td>
<td>2,7</td>
<td>12</td>
<td>2.29</td>
<td>-15</td>
</tr>
<tr>
<td>Code/Unit Test</td>
<td>15,6</td>
<td>70</td>
<td>13.37</td>
<td>-14</td>
</tr>
<tr>
<td>Integr./Syst. Test</td>
<td>2,5</td>
<td>11</td>
<td>2.1</td>
<td>-16</td>
</tr>
<tr>
<td>Total</td>
<td>22,2</td>
<td>100</td>
<td>19.1</td>
<td>-13</td>
</tr>
</tbody>
</table>

**Project C**

Project C employed object-oriented technology while much overtime had been spent by the designers and programmers of project C. The estimate was generated using ELYS-RT. Sensitivity analysis has been applied to ELYS-RT in order to verify that the weights of the various cost drivers in the cost model match what really happened during the project. Sensitivity analysis was performed on the basis of the answers of the project leader; the analysis revealed no big differences so no special calibration considerations were made. The deviation between the estimate and the actual effort (only -12.4%) confirms this conclusion. In the following table the results of the estimation of the effort and the related errors per life-cycle phase are reported.
<table>
<thead>
<tr>
<th>Life-cycle phase</th>
<th>Actual (staff Mths)</th>
<th>Nominal effort for each phase (%)</th>
<th>Estimate (staff Mths)</th>
<th>% Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reqs Anal.</td>
<td>101,17</td>
<td>37,5</td>
<td>96,78</td>
<td>-4,3</td>
</tr>
<tr>
<td>Prod. Design</td>
<td>42,52</td>
<td>17,5</td>
<td>45,17</td>
<td>6,2</td>
</tr>
<tr>
<td>Detailed Design</td>
<td>46,77</td>
<td>15,0</td>
<td>38,71</td>
<td>-17,2</td>
</tr>
<tr>
<td>Code/Unit Test</td>
<td>64,41</td>
<td>18,0</td>
<td>46,45</td>
<td>-27,5</td>
</tr>
<tr>
<td>Integration</td>
<td>16,09</td>
<td>2,0</td>
<td>5,16</td>
<td>-64,5</td>
</tr>
<tr>
<td>System Test</td>
<td>23,91</td>
<td>10,0</td>
<td>25,81</td>
<td>7,9</td>
</tr>
<tr>
<td>Total</td>
<td>294,87</td>
<td>100,0</td>
<td>258,08</td>
<td>-12,4</td>
</tr>
</tbody>
</table>

**Project D**

Project D was also developed using object oriented methodology; this project was estimated using ELYS-B. The project was estimated by breaking down the release into four sub-projects; the effort for the requirements and testing phases was calculated considering the project as a whole, while the effort for the design and coding phases was estimated by analyzing separately the four subprojects. Sensitivity analysis was also performed for project D. In the following table the results of the estimation of the effort and the related errors per life-cycle phase are reported.

<table>
<thead>
<tr>
<th>Life-cycle phase</th>
<th>Actual (staff Mths)</th>
<th>Nominal effort for each phase (%)</th>
<th>Estimate (staff Mths)</th>
<th>% Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reqs Anal.</td>
<td>20,00</td>
<td>5</td>
<td>15,59</td>
<td>-22</td>
</tr>
<tr>
<td>Arch. &amp; Spec</td>
<td>79,00</td>
<td>30</td>
<td>53,57</td>
<td>18</td>
</tr>
<tr>
<td>High Level Des.</td>
<td>33,00</td>
<td>5</td>
<td>15,59</td>
<td>-52</td>
</tr>
<tr>
<td>DD/Code/Unit Test</td>
<td>187,50</td>
<td>40</td>
<td>124,70</td>
<td>-33</td>
</tr>
<tr>
<td>Integr./Syst. Test</td>
<td>54,75</td>
<td>20</td>
<td>46,12</td>
<td>-15</td>
</tr>
<tr>
<td>Total</td>
<td>374,25</td>
<td>100</td>
<td>255,62</td>
<td>-20</td>
</tr>
</tbody>
</table>

**4. Conclusions**

The collaborative project with Bellcore has been extremely useful for CSELT. It provided the opportunity to become familiar with the estimation methodology and tools of our project partners and increased the CSELT experience in estimating software by working on programs produced in a different development environment in respect with the European reality.
5. Acknowledgements

The CSELT team would like to thank all the members of the Bellcore team and in particular Stuart Glickman and Bill Pitterman for their collaboration and their patience in coping with the every day problems of the project.

6. References

[BIS94] Bisio, R Stamelos, I.
'Software Cost Prediction by Case Based Reasoning',
CSELT Technical Report, Sept. 94

[CAP93] Capacci, C Stamelos, I.
'Constructing Software Cost Models'

[ELY] "ELYS, Estimation des couts de developpement des projets logiciels: Manuel d' utilisation"

[GLI93] Glickman, S Brown, R Pitterman, B Stamelos, I
'A Tool for Assessing Project Estimation Capability of a Software Development Organization'
Proc. 8th International Forum on COCOMO and Software Cost Modelling; (1993)

[ROS94] Rosenga, G Stamelos, I.
'On the Use of Neural Networks for Software Cost Estimation',
submitted to the Journal of Systems and Software

[STA94] Stamelos, I.
'Object Oriented Cost Estimation',
submitted to the Information and Software Technology journal