# PROJECT CONTROL: THE HUMAN FACTOR

Proceedings

of

# ESCOM - SCOPE 2000

18-20 April 2000
Munich, Germany

Combining the 11th ESCOM (European Software Control and Metrics Conference) Conference with the 3rd SCOPE (Software Certification Programme in Europe) Conference.

Editors


Katrina D. Maxwell
Datamax, France

Rob J. Kusters
Eindhoven University of Technology & Open University,
The Netherlands

Erik P.W.M. van Veenendaal
Improve Quality Services & Eindhoven University of Technology,
The Netherlands

Adrian J.C. Cowderoy
Nexusworld.net, United Kingdom

## ESCOM – SCOPE 2000 CONFERENCE OFFICERS

**General Chair**
Dieter Rombach (Fraunhofer IESE, Germany)
**ESCOM Programme Chair**
Katrina Maxwell (Datamax, France)
**SCOPE Programme Chair**
Erik van Veenendaal (Improve Quality Services & Eindhoven University of Technology, NL)
**Associate Programme Chair**
Rob. J. Kusters (Eindhoven University of Technology and The Open University, NL)
**Network Chair**
Adrian Cowderoy (Nexusworld.net, UK)
**Conference Manager**
Jo Cowderoy (Escom Conference Ltd., UK)

## ESCOM – SCOPE 2000 PROGRAMME COMMITTEE

Galtierro Bazzana, Onion, Italy
Miklos Biro, MTA SZTAKI, Hungary
Lionel Briand, Carleton University., Canada
A. Winsor Brown, USC, USA
Giovanni Cantone, University of Rome, Italy
Adrian Cowderoy, Nexusworld.net, UK
Gerard Duin, KEMA, The Netherlands
John Elliott, DERA, United Kingdom
Fred Heemstra, Open University, The Netherlands
John Jenkins, Middlesex University, United Kingdom
Ross Jeffrey, University of New South Wales, Australia
David Longstreet, Longstreet Consulting, USA
Serge Oligny, UQAM Software Engineering Management Research Laboratory, Canada
Michael O'Duffy, Centre for Software Engineering, Ireland
Shari Lawrence Pfleeger, Software/Systems Inc., USA
Loredana Mancini, O. Group, Italy
Roberto Meli, Data Processing Organisation, Italy
Peter Mellor, CSR, United Kingdom
Ingunn Myrtveit, The Norwegian School of Management, Norway
Domenico Natale, SOGEI, Italy
Martin Shepperd, Bournemouth University, United Kingdom
Erik Stensrud, Ernst&Young Management Consulting, Norway
Richard D. Stutzke, SAIC, USA
Jos Trienekens, Eindhoven University, The Netherlands
June Verner, Drexel University, USA
Claes Wohlin, Lund University, Sweden

**Conference Office**
Jo Cowderoy, ESCOM Conference Ltd
1 Walstead Cottages, Walstead, W.Sussex,
RH16 2QQ, UK.
Phone/fax +44(UK) 1444 484093.
Email: Jo.Cowderoy@escom.co.uk
**http://www.escom.co.uk/**

**ESCOM – SCOPE – 2000 SUPPORTERS**
CSR, The Centre for Software Reliability, UK
Fraunhofer IESE, Germany
Philotech, Germany
Eindhoven University of Technology, NL
Testnet, The Netherlands
Improve Quality Services, The Netherlands

## TABLE OF CONTENTS

# Planning the non-plannable maintenance[1]

Macario Polo, Mario Piattini, Francisco Ruiz

## Abstract

*In this paper we present a method for estimating the quantity of resources to be used for correcting software errors in a period, during the software maintenance process, in such a way that the organisation that maintains the software has no economical losses.*

*It is well known that software maintenance is the most costly stage of the software life cycle. So, it is important to provide techniques of this type, in order to decrease its high costs. Moreover, this is specially useful nowadays, when there is in Europe an excess of demand of skilled people in Information Technologies of about one million with respect to the offer.*

*The method is included as a technique within MANTEMA, a methodology for software maintenance.*

## 1. Introduction

MANTEMA is a methodology for managing the Software Maintenance Process which has been jointly developed by our university and Atos ODS, a multinational organisation which provides software maintenance services to big banking and industrial enterprises.

This methodology defines precisely and rigorously all the activities and tasks which must be executed during the Maintenance process, and explicitly considers the integration of the necessary activities for establishing and completing outsourcing relationships between customer and supplier organisations. Outsourcing of software services is a business activity which is experiencing a strong and growing. The following five types of maintenance are defined in MANTEMA:

1 Urgent-corrective
2 Non-urgent corrective
3 Perfective
4 Preventive
5 Adaptive

The four latest have been grouped into just one type (called "plannable maintenance"), since they share similar execution procedures. To maintain the same notation, we have called the urgent corrective "non-plannable maintenance", which could be mapped to the "emergency" type of maintenance defined in [6].

This type of maintenance really does not lend itself to planning , although an increase in corrective modification requests after every software release has been observed by different authors (Even, some authors have proposed a method (based upon the dynamic life model of May) for predicting the temporal distribution of the arrival of maintenance requests, including corrective ones.

In this paper we present the method proposed in MANTEMA to estimate the resources to be devoted to the non-plannable maintenance. The method uses some predictive technique of the future arrival of corrective maintenance requests, as well as the economical parameters of the project, in order to mathematically determine the quantity of human resources to be dedicated to corrective maintenance with no economical loss.

---

[1] This work is partially supported by the projects: *MANTIS* (CICYT/EU 1FD97-1608) and *MPM: Mejora del Proceso de Mantenimiento* (Ministerio de Industria y Energía, Iniciativa ATYCA, TA15/1999)

This method is especially suitable for organisations which supply outsourcing services of software maintenance, but it is applicable to any situation, whether there is or is not an outsourcing relationship: depending on the case in question, the model will take into account certain variables among the set of possible ones.

This is important for to saving costs in the maintenance process which, as it is well known, is the most costly stage of the software life cycle. Also it is very useful to know how many resources to dedicate to every project, taking into account that there were approximately 500.000 open IT positions in Western Europe in 1998, quantity that could grow up to 2.4 million positions by 2002 [2].

This paper is organised as follows: section 2 mentions some works which have been useful to build the model presented. In section 3, the method to plan resources for non-plannable maintenance is explained. An example is proposed and resolved in section 4. In section 5, we draw our conclusions.

## 2. Background research

Kung and Hsu have been maybe the most precise authors in the characterisation of the distribution of maintenance requests during the life of software system [8]. In fact, they distinguish the following four stages:

1 Introduction, characterised by the arrival of user support requests, since users do not yet know yet how to operate with the new system.
2 Growth, characterised by the increase in repair requests. This is motivated because users have become experts in the use of the system and they know and utilise all its details, and exploit all its characteristics.
3 Maturity, characterised by the arrival of enhancement requests, since users need more, greater and new utilities from the system.
4 Decline, characterised by the disappearance of any type of user requests and the desire for the replacement of the old system by a new one.

These distributions may be illustrated through a graphic which, in the ideal case, shows three bell-shaped lines, each one representing one type of user request. In the two real cases presented by the authors (see the reference), such lines are only slightly bell-shaped, although they show the highest peaks in their respective moments of Introduction, Growth and Maturity.

Calzolari et al. have used the data of repairs of Kung and Hsu to illustrate their method for estimating the future distribution of maintenance requests. They have adapted the classical predator/prey model of May, which models populations by means of differential equations. They map a new software release with an instantaneous introduction of errors in the system, and the corrective maintenance effort with a predator of such errors.

From these considerations, get a very approximated model of the future distribution of corrective maintenance requests.

## 3. Planning resources for non-plannable maintenance

In MANTEMA, one of the tasks which must be executed before beginning a *plannable* maintenance intervention is the estimation of the needed resources to serve the request. This estimation is relatively simple, since there is time enough to do it and there are several methods to calculate the effort of concrete maintenance interventions (for example: [7], [9], [14]).

But, on the other hand, it is quite difficult and of little use to make an estimation of the necessary resources to fix a critical program which requires imminent correction. However, in the case of *non-plannable* maintenance, we can use a predictive model of the future distribution of corrective maintenance requests (for example, that one of [3]) to allocate, during a period, the needed resources to serve all the requests.

### 3.1. Participant organisations

The method we are going to present in the next section is applicable to maintenance projects carried out in outsourcing or non-outsourcing environments. However, different models are obtained depending on its existence. In order to clarify both approaches, we will use the following terms to make reference to the participant organisations [12]:

- Maintenance organisation: is the organisation which supplies the maintenance service.
- Customer: is the organisation which owns the software and requires the maintenance service.

When there is an outsourcing relationship, each organisation is obviously independent from the other one. It is possible, however, that the same organisation owns and maintains its software: in this case, the single organisation will act sometimes in its role of *Maintenance organisation*, and sometimes as *Customer*.

### 3.2. Service level indicators

When there is outsourcing, both organisations covenant different *Service level indicators*, which are commonly accepted as the minimum quality levels with which the Maintenance organisation will supply its service. Among other more general Service level indicators [4], maintenance projects require the agreement of the following ones:

- *TRUC* (Resolution Time of Urgent Corrective modification requests): This is the maximum time that the maintenance organisation may employ in the solution of a critical anomaly (urgent-corrective) without being sanctioned.
- *TRNUC* (Resolution Time of Non-Urgent Corrective modification requests): This is the maximum time to resolve non-critical modification requests.
- The agreement of Service level indicators for the rest of plannable types of maintenance (especially perfective and adaptive) is not recommended without a carefulness previous study, since it is difficult to anticipate times and efforts for this kind of interventions, which may imply big additions of functionality. On the other hand, interventions of preventive maintenance are not usually covenanted, but a commitment to progressive preventive maintenance when corrective or perfective is being carried out (for example, the Maintenance organisation commit itself to decrease the mean cyclomatic complexity of all the modified modules).

When both organisations agree with the indicators, others must be set up as follows:

- *NMRTRUC*, which is the maximum Number of Modification Requests assumable in a period in the time *TRUC* (for example: 30 urgent-corrective modification requests per month, but with a maximum of 12 per week).
- *NMRTRNUC*, which is the maximum Number of Modification Requests assumable in a period in the time *TRNUC*.

### 3.3. Economic modelling

When there is outsourcing, it is normal that the Maintenance organisation suffers economical sanctions if the modification requests are not done within a specified time limit. However, this fact does not exempt the Customer from paying the resources to the

Maintenance organisation in such manner that a "bargain sale" is done with each payment. Also, the Customer suffers economical losses for each hour of system stop and the Maintenance organisation must pay sums of variable quantity depending on the amount of time devoted.

In order to prepare the economic model of the maintenance project, we must identify every one of these factors, within urgent-corrective maintenance:

- $\mu_C$ is the price (in euros, for example) that the Customer pays to the Maintenance organisation by each contracted hour of service.
- $\mu_{MO}$ is the price (in euros) that the Maintenance organisation pays to the resources. If there are no more economic parameters in the project, then it is obvious than $\mu_{MO}$ should be greater than $\mu_C$ if the Maintenance organisation wants to earn money.
- $s$ is the magnitude of the sanction that the Maintenance organisation must pay to the Customer for each hour of delay in the error correction.

If the Maintenance organisation dedicates less resources than needed, then it will incur delays and the respective sanctions. However, the Maintenance organisation may prefer to incur such delays if the sanctions plus costs of devoting less resources than needed is less than the cost of devoting all the needed resources. Let us study how many resources the Maintenance organisation must devote in order not to lose money.

Table 1 shows a generic situation of a Maintenance project for a period of $n$ days. With the exception of the last row, which shows some totals, the first column is the number of the day; the second one is the number of hours that the Maintenance organisation will devote to correct the errors (note that it plans $p$ hours for every day). The third column ($h_i$) is the predicted number of hours that will be needed to fix errors. Supposing that $p \leq h_i$, the fourth column shows the number of days that the Maintenance organisation will need to fix the errors occurred in the *i-th* day. The fifth column shows the day when the errors produced in the respective day will be corrected (for example, if $h_i=8$ $\forall i$ and $p=4$, errors of the first day will be corrected the second day, which implies to begin with delaying the correction of errors of the second day, and so on). The delay in the correction of the errors of each day is shown in the sixth column, and its respective sanction appears in the seventh and last column.

In the maintenance project of Table 1, the Maintenance organisation will pay daily $\mu_{MO}p$ euros to the contracted resources, whereas the Customer will pay daily $\mu_C h_i$ euros. During the $n$ days, these quantities will be respectively $C(p)=\mu_{MO}np$ and $\sum_{i=1}^{n}\mu_C h_i$. In order to simplify the problem, let us suppose that $h$ are the daily hours that the resource is needed (this does not disturb the model, since the Maintenance organisation is planning the $p$ daily hours for $n$ days, and then we can say that $h$ is the sum of all $hi$ divided into $n$). Taking this into account, the Maintenance organisation gets $I(p)=\mu_C\, nh$ euros from the Customer in all the days. Also we can rewrite the first $n$ terms of the last column in this way:

$$s\cdot\frac{h-p}{p};\ s\cdot\frac{2\cdot(h-p)}{p};\ s\cdot\frac{3\cdot(h-p)}{p};\ldots;\ s\cdot\frac{n\cdot(h-p)}{p}$$

*Table 1. Some parameters of a generic maintenance project.*

| Day | Planned hours | Needed hours | Needed days | Day of end | Delay | Sanction |
|---|---|---|---|---|---|---|
| 1 | $p$ | $h_1$ | $\frac{h_1}{p}$ | $\frac{h_1}{p}$ | $\frac{h_1}{p}-1=\frac{h_1-p}{p}$ | $s\cdot\left(\frac{h_1}{p}-1\right)=s\cdot\frac{h_1-p}{p}$ |
| 2 | $p$ | $h_2$ | $\frac{h_2}{p}$ | $\frac{h_1}{p}+\frac{h_2}{p}$ | $\frac{h_1+h_2-2p}{p}$ | $s\cdot\left(\frac{h_1+h_2}{p}-2\right)=s\cdot\frac{h_1+h_2-2p}{p}$ |
| 3 | $p$ | $h_3$ | $\frac{h_3}{p}$ | $\frac{h_1+h_2+h_3}{p}$ | $\frac{h_1+h_2+h_3-3p}{p}$ | $s\cdot\frac{h_1+h_2+h_3-3p}{p}$ |
| ..... | | | | | | |
| $n$ | $p$ | $h_n$ | $\frac{h_n}{p}$ | $\sum_{i=1}^{n}\frac{h_i}{p}$ | $\frac{\sum_{i=1}^{n}h_i-np}{p}$ | $s\cdot\frac{\sum_{i=1}^{n}h_i-np}{p}$ |
| | $pn$ | $\sum_{i=1}^{n}h_i$ | | | $\sum_{j=1}^{n}\frac{\sum_{i=1}^{j}(h_i-ip)}{p}$ | $s\cdot\sum_{j=1}^{n}\frac{\sum_{i=1}^{j}(h_i-ip)}{p}$ |

The last cell of the last column (total sanction, dependent on $p$, which is the daily quantity of resources to allocate by the Maintenance organisation) is the sum of these terms:

$$S(p)=s\cdot\frac{h-p}{p}+s\cdot\frac{2\cdot(h-p)}{p}+s\cdot\frac{3\cdot(h-p)}{p}+\ldots+s\cdot\frac{n\cdot(h-p)}{p}=$$

$$=s\cdot\frac{h-p}{p}\cdot(1+2+3+\ldots+n)=s\cdot\frac{h-p}{p}\cdot\sum_{i=1}^{n}i=s\cdot\frac{h-p}{p}\cdot\frac{n\cdot(n+1)}{2}=\frac{n\cdot(n+1)\cdot s\cdot(h-p)}{2p}$$

Now we have all the needed information to calculate $p$: the Maintenance organisation will use $p$ hours of resource ($p\leq h$) from the value which equals $I(p)$ with $C(p)+S(p)$. Let us resolve this equation:

$$I(p)=C(p)+S(p)$$

$$n\mu_C h=n\mu_{MO}p+\frac{n\cdot(n+1)\cdot s\cdot(h-p)}{2p}$$

$$2pn\mu_C h=2pn\mu_{MO}p+n\cdot(n+1)\cdot s\cdot(h-p)$$

$$2pn\mu_C h=2pn\mu_{MO}p+n\cdot(n+1)\cdot s\cdot h-n\cdot(n+1)\cdot sp$$

$$2p^2n\mu_{MO}-2pn\mu_C h+n\cdot(n+1)\cdot s\cdot h-n\cdot(n+1)\cdot sp=0$$

$$2p^2\mu_{MO}-2p\mu_C h+(n+1)\cdot s\cdot h-(n+1)\cdot sp=0$$

$$p^2 2\mu_{MO}-p[2\mu_C h+(n+1)s]+(n+1)\cdot s\cdot h=0 \quad \textbf{(1)}$$

If the Maintenance organisation resolves this equation with the parameters of its maintenance project, then it will know the quantity of resources to be devoted in order not to incur economical loss.

But this problem can also be resolved from the Customer's point of view: in fact, the Customer sees as a profit the payment of sanctions by the Maintenance organisation (function

*S)*, and has as costs the payment of resources (function *C)* and a variable loss which depends upon the duration of the error.

During the $n$ days, this variable loss will be:

$L(p)=\delta\cdot n\cdot h$, being $\delta$ the cost (in euros) for each hour that the system is stopped.

Therefore, the Customer will not loss money if the sanctions paid by the Maintenance organisation are greater than the variable loss plus the cost of resources. In other words:

$$L(p)+C(p)=S(p)$$

$$\delta\cdot n\cdot h+\mu_C\cdot n\cdot h=\frac{n\cdot(n+1)\cdot s\cdot(h-p)}{2p}$$

$$\delta\cdot h+\mu_C\cdot h=\frac{(n+1)\cdot s\cdot(h-p)}{2p}$$

$$2ph(\delta+\mu_C)=(n+1)\cdot s\cdot(h-p)$$

$$2\cdot p\cdot h(\delta+\mu_C)=(n+1)\cdot s\cdot h-(n+1)\cdot s\cdot p$$

$$2\cdot p\cdot h\cdot(\delta+\mu_C)+(n+1)\cdot s\cdot p=(n+1)\cdot s\cdot h$$

$$p\cdot[2\cdot h\cdot(\delta+\mu_C)+(n+1)\cdot s]=(n+1)\cdot s\cdot h$$

$$p=\frac{(n+1)\cdot s\cdot h}{2\cdot h\cdot(\delta+\mu_C)+(n+1)\cdot s} \quad \textbf{(2)}$$

We can add a last point of view for studying this problem: an organisation which maintains its own software. In this case, the organisation will not incur sanctions, but will suffer variable losses. Cost of an hour of resources will be simply $\mu$, and the daily cost of resources is $\mu p$, because the organisation pays the real dedication of the resources. The interesting point in this case is when cost of dedicating the needed resources is equal to cost of dedicating less resources than needed plus variable losses:

$$I(p)=L(p)+C(p)$$

$$n\cdot\mu\cdot h=n\cdot\delta\cdot h+n\cdot h\cdot p$$

$$\mu\cdot h=\delta\cdot h+h\cdot p$$

$$p=\frac{h\cdot(\mu-\delta)}{\mu} \quad \textbf{(3)}$$

## 4. Example

Let us imagine a maintenance project with two organisations involved: the Customer and the Maintenance organisation. This one wishes to know how many resources to dedicate on a daily basis for the next 30 days of a project. It has estimated 8 hours of needed resources per day. It pays 50 euros per hour for its resources, and receives 100 euros per hour from the Customer. Which in turn sanctions with 50 euros every hour of delay in the corrections.

I f we translate this problem into the terms used in the previous section, we obtain:

$n = 30$
$\mu_C = 100$ euros/hour
$\mu_{MO} = 50$ euros/hour
$s = 50$ euros/hour
$h = 8$ hours

Applying Eq. (1):

$$p^2 \cdot 2 \cdot 50 - p[2 \cdot 100 \cdot 8 + (30+1) \cdot 50] + (30+1) \cdot 50 \cdot 8 = 0$$

$$100 \cdot p^2 - 3150 \cdot p + 12400 = 0$$

$$p = \frac{3150 \pm \sqrt{3150^2 - 4 \cdot 100 \cdot 12400}}{2 \cdot 100} = \frac{3150 \pm 2227.67}{200}$$

$$p_1 = 26.88 \text{ hours}; \; p_2 = 4.61 \text{ hours}$$

Obviously, to use more than 8 hours per day has no sense, since then there will be an excess of resources and the Maintenance organisation will lose money. Then, $p_1$ cannot be used for this problem. Table 2 shows the evolution of comings in (function *I*), costs of resources (function *C*) and amount of sanctions (function *S*). The fourth function (*B*) the benefits is calculated in this way: $B=I-(C+S)$.

*Table 2. Economical parameters according to p.*

| p | I(p) | C(p) | S(p) | B(p) |
|---|---|---|---|---|
| 1 | 24000 | 1500 | 162750 | -140250 |
| 2 | 24000 | 3000 | 69750 | -48750 |
| 3 | 24000 | 4500 | 38750 | -19250 |
| 4 | 24000 | 6000 | 23250 | -5250 |
| 4,61 | 24000 | 6915 | 17097,0716 | -12,0715835 |
| 4,62 | 24000 | 6930 | 17009,7403 | 60,2597403 |
| 5 | 24000 | 7500 | 13950 | 2550 |
| 6 | 24000 | 9000 | 7750 | 7250 |
| 7 | 24000 | 10500 | 3321,42857 | 10178,5714 |
| 8 | 24000 | 12000 | 0 | 12000 |
| 9 | 24000 | 13500 | 0 | 10500 |
| 10 | 24000 | 15000 | 0 | 9000 |

Note that the use of more than approximately 4.61 resources allows the Maintenance organisation to obtain benefits.

Let us study the problem from the Customer's point of view. In this case, let be the variable loss be: $\delta$=120 euros/hour. Applying Eq. (2):

$$p = \frac{(30+1)\cdot 50\cdot 8}{2\cdot 8\cdot (120+100)+(30+1)\cdot 50} = \frac{12400}{5070} = 2.45 \text{ hours}$$

This means that if the Maintenance organisation devotes less than 2.45 hours of resource in the 30 days, then the Customer will earn money thanks to the sanctions imposed. However, this result is very dangerous, as we can also deduce from Figure 1: when the Maintenance organisation devotes more sources, the Customer loses more money. It seems that the ideal case is that the Maintenance organisation devotes 0 hours of resource to earn a lot of money from sanctions. However, risks of this kind of decisions should be carefully studied by the Customer, because new indirect costs (not taken into account until now) would appear. In any case, the Maintenance organisation would not allow that quantity of resources, since it is not in the interests of either of the organisations.

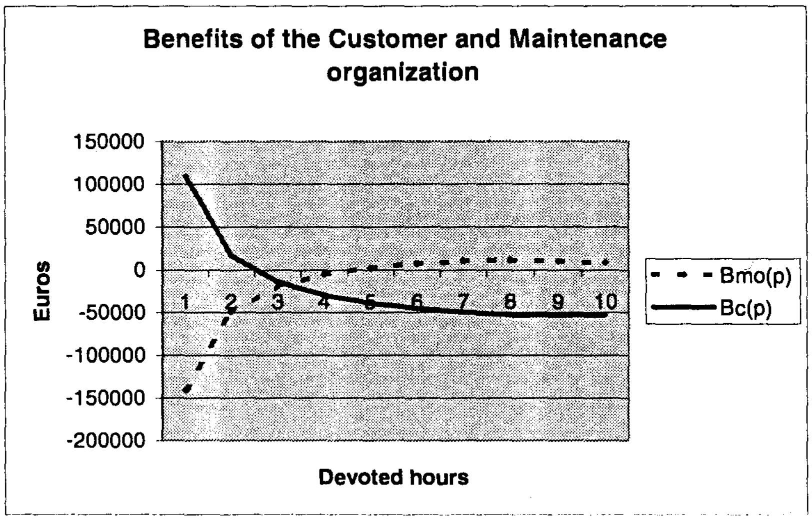


*Figure 1. Benefits of both organisations, depending on p.*

## 5. Discussions and model extensions

In spite of the example has shown that the Customer would obtain profits if the Maintenance organisation devotes very little resources, the reality is that very big delays may imply unacceptable production losses. This means that there are other costs, not directly considered in this model. A similar effect occurs from the Maintenance organisation point of view, since the little dedication may imply others risks and costs, as the cancellation of the maintenance contract. This fact may be included in the model considering, as an additional cost, a percentage of the losses suffered by the Customer, well linearly or exponentially. For example:

$$C(p) = n\cdot \mu_{MO}\cdot p + k\cdot (\delta\cdot n\cdot h)^{1\pm\varepsilon}$$

## 6. Conclusions

In this paper we have shown a method for estimating the quantity of resources to be devoted to maintenance projects, in such a way that this quantity does not originate economical losses in the Maintenance organisation.

The results of this method are interesting for Maintenance organisations, since they can have different maintenance projects and not enough resources to serve all of them adequately. This is a usual situation at this time, when in Europe there is a demand for programmers which exceeds the offer very much.

The method can also be applied from the Customer's point of view, although in this case the results obtained may involve high risks that the Customer should analyse carefully.

Also it can be applied by organisations which maintain its own software: in this case, they do not have sanctions and the cost of resources is directly paid by them, without intermediaries.

## 7. References


[1] Basili, V., Briand, L., Condon, S., Kim, Y., Melo, W. and Valett, J.D., "Understanding and Predicting the Process of Software Maintenance Releases". Proceedings of the International Conference on Software Engineering. IEEE, 1996.

[2] Bourke, T.M. (1999). "Seven major ICT companies join the European Commission to work towards closing the skills gap in Europe" (Press Release), in Proceedings of the New Partnerships to Close Europe's Information and Communication Technologies Skills Gap. Available also at (Jan., 20, 2000): http://www.career-space.com/whats_new/press_rel.doc

[3] Calzolari, F., Tonella, P. and Antoniol, G., "Modelling Maintenance Effort by Means of Dynamic Systems". Proceedings of the Third European Conference on Software Maintenance and Reengineering. Amsterdam (The Netherlands), IEEE Computer Society, Los Alamitos, CA, USA, 1998.

[4] De Vogel, M., "Outsourcing and metrics". Proceedings of the 2nd European Measurement Conference, FESMA'99. Federation of European Software Metrics Association/Technologisch Instituut, 1999.

[5] Hoffman, T., "Users say move quickly when outsourcing your personnel". Computer World, march 1997, p.77

[6] IEEE, "IEEE Std 1219-1992, Standard for Software Maintenance". The Institute of Electrical and Electronics Engineers, Inc., NY:EE.UU, 1992.

[7] Jorgensen, M., "Experience with the Accuracy of Software Maintenance Task Effort Prediction Models". IEEE Transactions on Software Engineering, 21(8), 1995, pp. 674-681.

[8] Kung, H.-J. and Hsu, Ch., "Software Maintenance Life Cycle Model" Proceedings of the International Conference on Software Maintenance. IEEE Computer Society, Los Alamitos, CA, USA, 1998.

[9] Niessink, F. and van Vliet, H., "Predicting Maintenance Effort with Function Points". Proc. of the International Conference on Software Maintenance. IEEE Computer Society, Los Alamitos, CA, USA, 1997.

[10] Pigoski, T. M., "Practical Software Maintenance. Best Practices for Managing Your Investment". John Wiley & Sons, USA, 1997.

[11] Polo, M., Piattini, M., Ruiz, F. and Calero, C., "MANTEMA: a complete rigorous methodology for supporting maintenance based on the ISO/IEC 12207 Standard". Proceedings of the Third European Conference on Software Maintenance and Reengineering. IEEE Computer Society, Los Alamitos, California (USA), 1999.

[12] Polo, M., Piattini, M., Ruiz, F and Calero, C., "Roles in the Maintenance Process". Software Engineering Notes, 24(4), 1999, pp. 84-86.

[13] Rao, H.R., Nam, K. and Chaudhury, A., "Information Systems Outsourcing". Communications of the ACM, 39(7), 1996, pp 27-28.

[14] Schneidewind, N.F., "How to evaluate legacy systems maintenance". IEEE Software, July/August 1998, pp. 34-42.

[15] Shepper, M., Schofield, C. and Kitchenham, B., "Effort estimation using analogy". Proceedings of the 18th International Conference on Software Maintenance. IEEE Computer Society Press, Los Alamitos CA, 1996.