

IT PROJECT METRICS

Ion IVAN¹

PhD, University Professor, Department of Economic Informatics
University of Economics, Bucharest, Romania
Author of more than 25 books and over 75 journal articles in the field of software quality management, software metrics and informatics audit. His work focuses on the analysis of quality of software applications.
E-mail: ionivan@ase.ro , **Web page:** <http://www.ionivan.ro>



Adrian VISOIU²

PhD Candidate, Assistant Lecturer, Economic Informatics Department,
University of Economics, Bucharest, Romania

E-mail: adrian.visoiu@csie.ase.ro



Dragos PALAGHITA³

4th year student, University of Economics, Bucharest, Romania

E-mail: dpalaghita@gmail.com



Abstract: *The objectives of IT projects are presented. The quality requirements that these projects must fulfill are established. Quality and evaluation indicators for running IT projects are built and verified. Project quality characteristics are presented and discussed. Model refinement for IT project metrics is treated and a software structure is proposed. For an IT project which is designed for software development, quality evaluation and project implementation mode metrics are used.*

Key words: *IT metrics; IT projects; quality; quality characteristics*

1. IT projects

They represent an important category of projects and their management implies specific particularities because:

- Their unique character; informatics systems and software are developed only one time to bring in new solutions;
- The reproduction of the resulted products is defined by costs nearing zero, determined by the possibility to make copies on modern storage devices;
- The work force used to develop the IT project is highly qualified and represents, in most cases, over 85% of the projects value;
- The development cycle of an IT project must include without question solution improvement stages to bring in the latest technologies in the field; neglecting these new technologies will lead to an old solution which may be rejected by the beneficiaries and the informatics market;

- The structure of the development cycle is adapted the pursued objective, the allocated resources, which leads to decisions regarding the continuance or abandonment of the projects, when the reviewed efficiency criterion suggest one of the two choices; this explains the fact that of 100 projects, over 50 are abandoned in the development process, only 10 projects get in maintenance and only 2-3 go in to the reengineering process;
- In the context of an evolving information and knowledge based society, all IT projects must be oriented to be user friendly, in order to grant free access to broader categories of information to every citizen.

IT project imply vast amounts of money. For projects in constructions, the amount of money needed range between thousands and millions of euros, and the resulting product is tangible, it becomes functional and is directly perceived.

For IT projects the amount of money needed starts from ten thousands euros and finally the result can only be seen in computers and networks, which are no more than 15% of the projects value. The hidden part of 85% must prove its efficiency through functionality, in time. This is why it is necessary building IT project metrics, to measure:

- How are the resources used;
- Which are the products' performances;
- How efficient will the products be in exploitation;
- Which are the risks of not finishing the project or to compensate deviations from the project.

Currently there are numerous metrics for every software typology, for every quality characteristic and for cost estimation.

It is important these metrics are analyzed and a practical approach for using quantitative methods is developed such that in the IT field the grounds of decisions will change the structure of projects and make the number of projects that get reengineered to increase.

2. Quality characteristics of IT projects

The IT projects quality characteristics form a dynamical, complex and optimal system.

The systemic character is given by the interdependences between characteristics. At a given time t , a quality characteristic of a project positively influences the level of another characteristic at time $t+1$. Also, that characteristic has a negative influence on the level of another quality characteristic.

The dynamic character is linked to the fact that interdependence intensity between characteristics varies in time. More, the importance of quality characteristics changes from one IT project to another, from user to user or even inside the project, the modification occurs from one stage of the project to another.

The complex character resides in the large number of quality characteristics, in the number of interdependencies, and mostly in the directions in which they manifest. The accumulated experience in elaborating and implementing projects shows that the characteristics system becomes more complex as the objectives are better pursued and the level of performance is higher.

The optimality of the quality characteristics system oversees the quantitative side of costs that the tendency to plan and complete projects with a higher quality level generates.

There is a solution of the quality optimization model that shows the moment in which the improvement of quality is no longer sustained through costs, which are far greater than the costs needed to fix the corresponding defects of a lower quality level of the IT project.

If M is considered the number of the quality characteristics C_1, C_2, \dots, C_M of the projects, for each characteristic must be stated:

- the content ;
- the factors of influence;
- the associated models
- the use of indicators in taking decisions;
- the indicators aggregation.

Complexity is the characteristic used in differentiating projects. A project A_i is more complex than a project A_j if:

- the type number of resources used is greater;
- the number of stages through which the project is finished is greater;
- the number of machines used is greater;
- the number of different operations done by each machine is greater;
- the qualification needed by workers to operate the machines is higher;
- the number of links between stages is greater;
- the execution time is greater;
- the organizational effort is greater;
- the density of execution procedures is higher.

A graph is associated to the project A_i in which the nodes are either stages or activities, and the arcs are the ones that set the precedence. If:

- n_1^i – the number of nodes of the graph $G(A_i)$ for the project A_i
- n_2^i – the number of arcs which link the nodes of the graph $G(A_i)$

the complexity CP of the project A_i in a Halstead way is given by:

$$CP(A_i) = n_1^i \log_2 n_1^i + n_2^i \log_2 n_2^i$$

A project A_i is more complex than a project A_j if and only if:

$$CP(A_i) > CP(A_j)$$

Meaning

$$\frac{(n_1^i)^{n_1^i} \cdot (n_2^i)^{n_2^i}}{(n_1^j)^{n_1^j} \cdot (n_2^j)^{n_2^j}} > 1$$

If in time in an organization projects are implemented of which a project A_k is considered a model, the relative complexities of the projects $CR(A_1), CR(A_2), \dots, CR(A_M)$ are obtained from:

$$CR(A_i) = \frac{C(A_i)}{C(A_k)}$$

The project model has the relative complexity $CR(A_k) = 1$.

The K completeness of project K is a quality characteristic which refers to two aspects:

K_o – the completeness of the offer regarding the content of the document which is under evaluation

K_i – the completeness of the implementation regarding the stages developing process and the completeness of the process

If p_1 is the percentage of the elaboration stage of the offer and p_2 is the percentage of the developing of the project $p_1 + p_2 = 1$ and $p_1, p_2 \in (0,1)$ it results that the level of completeness KP of project A_i is given by:

$$KP = p_1 * K_o + p_2 * K_i.$$

The level of completeness K of any construction regards:

np – the number of parts imposed to consider the construction accepted

nr – the number of parts effectively realized

and the relation is obtained:

$$K = \frac{\min\{np, nr\}}{\max\{np, nr\}}$$

A project contains:

- eligibility criterion to which variables that belong to $\{0,1\}$ are associated to; if in the project the imposed requirements are respected the value 1 is given to a criterion Cr_i ; if a project must fulfill L eligibility criterions the indicator $IL = \prod_{i=1}^L \alpha_i$ is calculated where α_i is the level of the Cr_i criterion, $\alpha_i \in \{0,1\}$;
- the technical side definition, in which definitions are given, processes are described, models are built; the completeness refers to the degree in which these descriptions include all the components known by specialists; the descriptions must offer a suggestive picture in connection with what is going on in the actual choused domain; there must not be a lack of elements which by their absence:
 - o prove that the authors do not have a grasp on the domain;
 - o show that the authors of the project do not have knowledge of materials, equipment, processes, operations, effects and models.

The specialists that analyze a new project have a clear image about the structure that includes mandatory parts, minimum requirements and by comparison see what is missing or what has been treated shallow.

Clarity is a very important characteristic not only for the offer made for a project but also to the development process.

The offer is a text. The implemented project is a product, a service, an action.

The clarity of text T_i associated with project A_i is analyzed in the following manner:

- the text T_i is decomposed in parts $a_{i1}, a_{i2}, \dots, a_{ir}$;
- each part a_{ij} $j = 1, 2, \dots, r$, is connected with resources, actions and equipments;
- the logic content of a_{ij} with the logic of the process development;
- parts of the text are connected with resources building up pairs (a_{ij}, r_p)

If the pairs parts of text – resources are completely defined then the text T is clear. If parts of text or resources are not paired, the project does not have a good clarity level. The clarity is influenced in a negative way by pairs in which for different parts of text the same resources come up.

The necessity of defining probabilities which are associated with the pairs (a_{ij}, r_p) also has a negative influence on clarity.

Correctness is quality characteristic of a project, through which the ones that make the offer ant participate in the development of the project ensure:

- the concordance between theory and practice
- the use of results obtained by other specialists, results that were verified in practice;
- the use of concepts as they were defined, maintaining the context unaltered;
- respecting the defined procedures for executing operations;
- through all the means used, that the requirements imposed by obtaining a high level of quality for the product or service in its final form, are respected.

The correctness of a project offer is demonstrated, and the correctness of a product or the execution of a service is pointed out either by analysis, or by establishing the effects the product or service will generate. A product or a service is correctly or incorrectly realized.

If it is accepted that the product or service is partially correct, the correctness measuring indicator moves from the values 0 or 1 to the interval $[0,1]$.

The problem of correctness in calculating costs and efficiency regards:

- the hypothesis on the basis of which the efficiency calculations are made;
- establishing the expenses levels for each chapter by pointing out quantities, unitary price, and interval limitation;
- using the calculation models as they were defined;
- bounding the variables regarding specific consumes through comparative analysis with projects already implemented;
- respecting the control keys;
- respecting the inequalities regarding the expense structure in conditions of the impossibility of not transmitting from one capital to another of the expenses.

For the definition of IT projects the following are defined:

- the criterions Cr_1, Cr_2, \dots, Cr_h ;
- the domains of variation measured for each criterion $[a_1, b_1], [a_2, b_2], \dots, [a_h, b_h]$.

If for criterion Cr_i the level x_i is measured, then the correctness of the project is given by:

$$CG = \prod_{i=1}^h x_i$$

$$\prod_{i=1}^h a_i \leq CG \leq \prod_{i=1}^h b_i.$$

The correctness norm is realized by defining the relation:

$$\prod_{i=1}^h \frac{a_i}{b_i} \leq CGN = \prod_{i=1}^h \frac{x_i}{b_i} \leq 1$$

There are numerous cases in which the relative correctness of a criterion CR_i is obtained using a report like:

$$CR_i = \frac{A_i}{h_i}$$

Where:

A – the number of fulfilled sub criteria

h – the total number of subcriteria

In this case the global correctness is

$$CG = \left(\prod_{i=1}^h CR_i \right)^{\frac{1}{h}}$$

Consistency is a quality characteristic which is used to point out to what extent an IT project is built respecting precise rules, without including components which have the role of annihilating what was prior executed.

The text of the project includes stages, chapters developed in ascendant order with the next components being constructed based on the ones before.

A project is under defined if the text or the product doesn't contain those elements that insure its functionality.

The project is over defined if it includes a lot more components than necessary. The numerous details make it hard to accomplish, and the function for which it is constructed is lost among many other functions, some having nothing in common with the initial objective.

Only a defined project, in which every component has its role well established, is a consistent project.

A project structure, SP , is considered, having the complexity $C(SP)$, structure which corresponds to projects, products or IT services considered to be functional, which exist and have a degree of satisfaction at user level greater than 0.8, the degree of satisfaction is given by:

$$GS = NF/NT$$

Where

NF – the number of users that have successfully finished the problem for which they used the resources of the project;

NT – the total number of users that accessed the resources belonging to the project.

The consistency CO for the project A_i is given by:

$$CO = \frac{\min\{C(SP), C(A_i)\}}{\max\{C(SP), C(A_i)\}}$$

Ability to complete is another characteristic that all evaluators take in consideration and it is influenced by:

- the experience of the team which proposes a solution for an IT problem;
- the estimation of resources used for implementing the solution;
- the estimated duration of the project;
- identified resources and procedures used to attenuate the effects.

The practice of elaborating an offer and implementing it presume:

- knowledge of technologies;
- rigorous knowledge of duration and minimal and maximal consumptions necessary to complete activities;
- the correct defining of precedence;
- identifying the operations needed to prepare;
- building a timeline for activities;
- regrouping the activities such that the equipment and highly qualified workforce are efficiently used.

If in a project are identified:

NA – number of activities;

D_i – time to complete activity i ;

K_i – number of resources necessary for completing activity i ;

x_{ij} – resource consumption j for completing activity i ;

t_{si} – start time for activity i ;

t_{fi} – end time for activity i ;

The planned and effective levels are established.

For the planned levels mark p will be used; the planned level for activity xxx will be xxx^p .

For the effective level the mark e will be used, such that variable xxx will have the effective level xxx^e .

The indicator IUR regarding the use of resources is built using the relation:

$$IUR = \frac{\sum_{i=1}^{NA} \sum_{j=1}^{ki} \alpha_{ij}}{\sum_{i=1}^{NA} k_i}$$

where

$$\alpha_{ij} = \begin{cases} 1, & x_{ij}^p = x_{ij}^e \\ 0, & \text{for the rest} \end{cases}$$

The ability to complete is seen also compared to optimistic or pessimistic attitude manifested by those who make an offer or implement a project.

If the planned level for a resource is x_{ij}^p , the minimal consumption for completing operation x_{ij}^{min} , and the maximal consumption for a resource is x_{ij}^{max} the level realistic approach is built as the indicator IR given by:

$$IR = \frac{\sum_{i=1}^{NA} \sum_{j=1}^{ki} \beta_{ij}}{\sum_{i=1}^{NA} k_i}$$

Where

$$\beta_{ij} = \begin{cases} 0, x_{ij}^p = x_{ij}^{\min} \text{ sau } x_{ij}^p = x_{ij}^{\max} \\ 1, \text{ for the rest} \end{cases}$$

Reliability is quality characteristic of IT projects which states the degree in which the stages in the life cycle of the project are developing successfully

The life cycle of an IT project includes:

- E1- the offer elaboration stage
- E2- project implementation stage
- E3- the current use of the product resulted after... through the project
- E4- the maintenance process
- E5- using the product post maintenance
- E6- taking the product out of use

Each of these stages has associated an importance coefficient $p_i, i=\{1,2,3,4,5,6\}$.

Each stage E_i is characterized by a duration $D(E_i)$.

The physical duration norm of the life cycle of the project is given by:

$$DFN = \sum_{i=1}^6 D(E_i)$$

The corrected duration norm of the project is given by:

$$DFNC = \sum_{i=1}^6 p_i D(E_i)$$

$$DFNC < DFN \quad \text{because } p_i \in (0,1) \sum_{i=1}^6 p_i = 1 \text{ resulting that } D(E_i) > p_i D(E_i) \text{ for } i=1,2,3,4,5,6.$$

In the background of every stage processes are running. If a process θ_j is stopped because of various causes, the duration of the process $DP(\theta_j)$ increases with $\Delta(\theta_j')$ which increase depends on:

- the complexity of the cause that determined the interruption;
- the ability of the team to eliminate the cause;
- the resources that are redirected to eliminate the cause;
- the place where the cause was spotted; as a cause is spotted later, the resources used and durations are increasing;

The physical running cycle norm of the project is identical with the effective duration if and only if $\Delta(\theta_j)=0 \forall j \in N$. In reality, throughout the lifetime there are recorded interruptions of specific stages $\Delta(E_i)$, such that the effective duration of the cycle:

$$DFE = \sum_{i=1}^n (D(E_i) + \Delta(E_i))$$

The norm reliability of the project $P, f(P)$ is given by:

$$f(P) = \frac{DFN}{DFE}$$

The corrected reliability $fc(p)$ is given by the relation:

$$f_c(P) = \frac{DFNC}{DFEC}$$

where $DFEC = \sum_{i=1}^n [p_i D(E_i) + h_i \Delta(E_i)]$ where $h_i, i=1,2,3,4,5,6$ are the fractions associated to the importance which is given to the duration of process delaying from the stages $E_i, i=1,2,..6$.

Compared to other products or services in which reliability is characterized by quality in use, at the beneficiary, giving the degree in which the product or service accomplishes the set objective, in case of the IT projects, the reliability of a project is mistaken with the reliability of the life cycle because:

- in all the stages the workforce plays an important role, representing more than 85% of the production costs;
- the successful functioning of the finite product strictly depends on the operating of each subsystem; the offer lays down the conceptual foundation and resource allocation; the elaboration of the offer is a stage of the life cycle of the IT project; the offer elaboration process reliability is essential to the entire product;
- the eventual deploying of an reengineering process is meant to redefine basic elements of the offer, and the multiple mobilization effect of the new structure of an IT project is limited to all the other stages that follow.

It is important to identify in the existing projects in the databases those variables with the help of which the levels of difference $\Delta(E_i)$ are calculated to help compute the effective reliabilities of these projects in the norm version and most of all in the corrected version.

Given the Project P in the table 1 below,

Table 1. Data for P project

Name	Norm duration	Effective duration	Interruption duration	Total
E1	6	12	8	20
E2	24	38	12	50
E3	50	60	40	100
E4	10	15	5	20
E5	5	10	10	20
E6	5	10	10	20
Total	100	145	85	230

The reliability of project P = $\frac{145}{230}$.

There are other quality characteristics of an IT project like:

- comparability
- reusability
- orthogonality
- reproduction ability

For these characteristics the research must be continued such that the results will be operational in daily practice.

3. IT project model quality refinement

Models are built starting from a model of hypothesis. It is necessary to go on to the stage of refining the models. The refinement process consists of:

- Reducing the number of variables;
- Decreasing as much as possible the degree of incorporated nonlinearity ;
- Guaranteeing minimal loss of information by using simplified models.

The refinement of IT projects assumes:

- Taking in consideration only the data that is collected usually for elaborating the offer and running the projects;
- Retaining in the indicator structure the variables that decisively influence the evolution of the project from a practical point of view;

Refinement is made on two levels:

- At the first level, from the set of indicators I_1, I_2, \dots, I_M a subset is retained formed by the indicators $I_{i_1}, I_{i_2}, \dots, I_{i_K}$, $K \ll M$.
- At the second level, the retained indicators operate with the variables $X_{i_1}, X_{i_2}, \dots, X_{i_H}$, $H \ll L$ instead of using the variables X_1, X_2, \dots, X_L .

It means that instead of analyzing M characteristics, a number of K characteristics is analyzed.

Instead of L variables, the refined indicators operate with H variables.

The effort must be reduced without significant loss of information obtained from using the metrics. The decisions taken based on the refined metrics of the IT projects must lead to maintaining the process of implementation in the limits recognized acceptability.

IT projects metrics refinement must allow the use of existent data bases, and the implementation of simplified indicators must not be affected by the defined data structures, without requiring new data structures or data reorganization.

The IT project metrics refinement process is carried on like in figure 1.

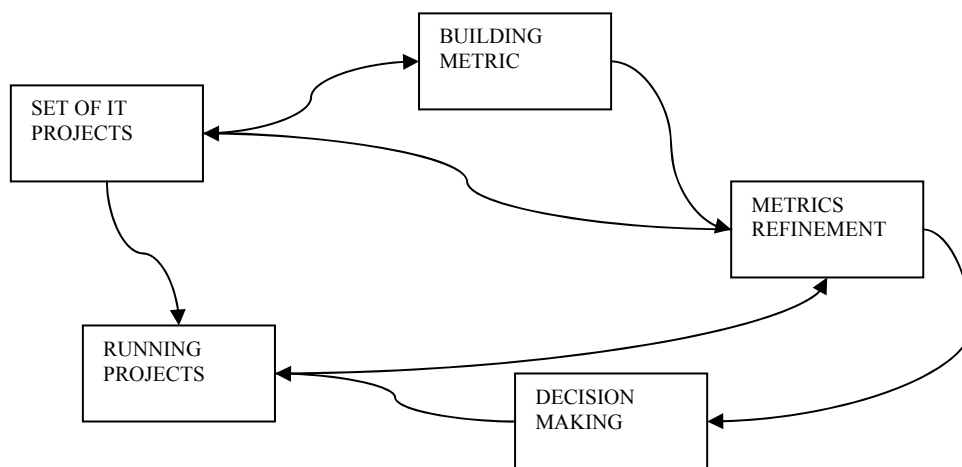


Figure 1. Usage of refined models

The refinement process is customizable by defining limits referring to:

- the minimum list of variables that defines the indicators;
- the accepted loss of information level.

Under the conditions the indicators I_1, I_2, \dots, I_K are considered in which the variables X_1, X_2, \dots, X_L are used, through refinement it is imposed:

- the maximum number of indicators to be M_{max} ;
- the maximum number of variables to be H_{max} ;
- the error level must not be higher than E .

Refinement presumes:

P1- definition of the initial solution for the refinement process which includes all the K indicators, all the L variables and the criterion through the measurements are made;

P2- taking the data bases that regard IT projects that have frequent definition classes and best implementation, known as successful projects.

P3- evaluation of indicators I_1, I_2, \dots, I_k with the data taken from the data bases for the L variables

P4- calculating the aggregated error criterion

P5- elimination of the x_i variable from the indicators

P6- evaluation of the indicators

P7- calculation of the criterion

P8- building the Er_i array

P9- repeating the steps $P_5 - P_8$ to eliminate the variable x_{i+1} from the list

P10- after building the array Er_1, Er_2, \dots, Er_L it is sorted in descending order and the variables for which the error has the smallest value are eliminated

P11- the steps $P_5 - P_{10}$ are repeated until the given error level is obtained; it is possible this level is reached in the first iteration and the refinement process stops;

Once the variable list is refined, the refinement of indicators starts.

P12- the list I_1, I_2, \dots, I_K of indicators and the list of refined variables X_1, X_2, \dots, X_H are considered;

P13- the indicators are calculated using the relation and are considered an initial solution for the aggregate performance indicator

$$IP^0 = \sum_{j=1}^H p_j I_j \quad \text{where } p_j - \text{weight}$$

and a certain level is obtained

P14- the indicator $I_k, k=1, \dots$ is eliminated

P15- IP is calculated and a vector $\alpha_1, \alpha_2, \dots$ is built

P16- steps $P_{14} - P_{15}$ are repeated until the indicators are eliminated one by one

P17- $\alpha_1, \alpha_2, \dots$ are ordered decreasing

P18- the indicator which has the smallest error difference with IP^0 is eliminated

P19- the steps $P_{14} - P_{18}$ are repeated until the maximum admitted number for indicators is reached

The refinement activity is a repeating process convergent to a stable process. To complete the refinement simulation methods and genetic algorithms and neuronal networks are built.

4. Software for implementing IT metrics

The software product for evaluating IT projects assumes:

- building a list with K quality characteristics $KAL_1, KAL_2, \dots, KAL_k$
- building a list of indicators I_1, I_2, \dots, I_k
- building a list with M variables
- making the connection between the M variables and fields in the data base
- launching the refinement process
- obtaining a sublist of characteristics with a sublist of indicators which are obtained with a sublist of variables
- validating the sublist of indicators as a metric of IT projects

The software product designed to implement IT metrics is a complex construction which:

- must be accessed off the internet;
- has user authentication ;
- accepts definitions of data bases;
- takes fields from the data base and constitutes the series that make estimations;
- allows the selection of structures of initial indicators I_1, I_2, \dots, I_k ;
- offers a list of performance criterion for selecting indicators and variables;
- generate indicator structures for the refinement process;
- calculates the level of refined indicators on request or periodically to fundament decisions.

The software product has the functions:

- user authentication;
- the management of data sets that correspond to the problems to solve;
- estimation of model coefficients;
- model refinement;
- model structure generation;
- aggregation level calculation to fundament decisions;
- options, parameters and solutions are saved if the data sets are valid;
- adding of information and identification data.

The software structure is given in figure 2.

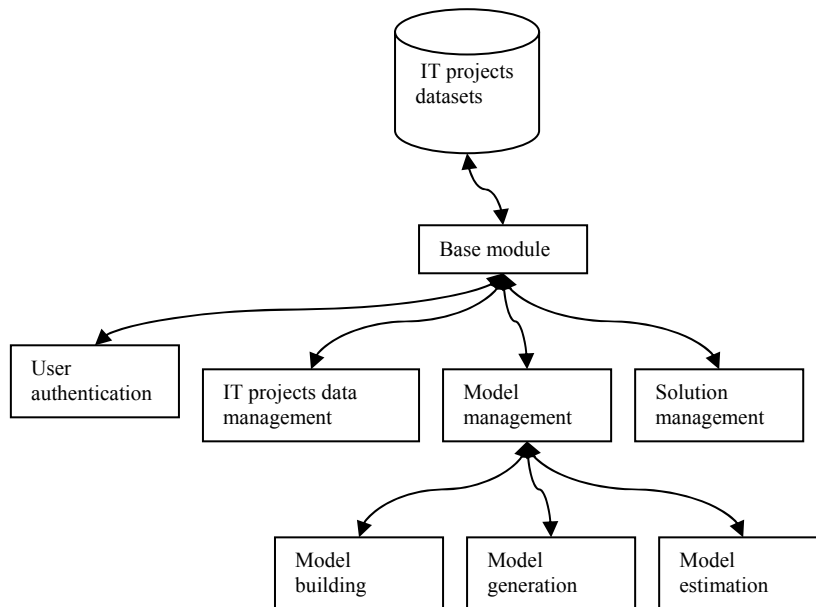


Figure 2. Software structure for IT project metrics implementation

The software product is taking numerous elements that constitute a modelbase.

The modelbase is a complex construction in which lists are included with:

- dependent variables and independent variables;
- linear and nonlinear models structures;
- registered data sets and generated data sets;
- implementation of the coefficients estimation algorithms;
- procedures for the hypothesis verification;
- procedures for the estimated values calculation;
- models hierarchizing procedures.

These very important components are elements with which the modelbase is populated. For the modelbase to become operational, an administration system has to exist.

First of all, the administration system has to operate distinctly with data sets, with the procedures and with the models structures.

Second of all, the administration system functions have to effectuate the fast finding of the data sets, of the models structures and of the procedures in order to secure the processes development in concordance with the demands of the analyst economist.

Third of all, the administration system has to be equipped with functions which permit the data sets adding, the models adding and the procedures adding. The perspective has to be changed, through which the modelbase bringing up to date implied data / models / procedures deleting or changing of some parts of these with new sequences.

The acceptance of the bringing up to date function exclusive through adding comes to bring a concordance between the natural way of understanding the evolution with the corresponding reflection of it on informatics level.

The modelbase administration system operates with non homogenous entities, important aspect in securing the flows consistency.

Fourth of all, an open character is secured, the users having at their disposal the possibility of defining personal algorithms for interpolation and extrapolation, for pseudoaleatory numbers generation, for the coefficients estimation, for implementing personal models selection criteria.

Fifth of all the, defining of the specific concepts regarding the finding, selection, extraction, targets the triplets (data, models structures, procedures), which group complex proceedings.

Sixth of all, a growth of the generalization degree for the transaction concept is produced, which in case of the modelbase implies the traversing of some flows in which it is operated simultaneously with data sets, with data structures and with procedures.

The new conglomerate, more complex than the object structure that includes the operands and operators, develops a new projection on the philosophy of designing an administration system, the administration system of the modelbase, in which new typologies specific to the implemented processes in the field of artificial intelligence are included besides the already usual proceedings.

Utilizations of the modelbase imply activations of some sequences of procedures from the modelbase.

The administration system of the modelbase is a construction with a very high complexity degree. The users must have the possibility of starting a small diversity of economic analysis projects.

For example, the coefficients estimation of a model on the basis of a data set consists of:

- the specification of the exogenous variables number and of the endogenous variables number;
- the specification of the data series terms number;
- the insertion of the data table;
- the delimitation of the variables list corresponding to the data series position.

Also, some results regarding the estimations quality are displayed.

This procedure is specific to the situation in which the user has a clear image on the phenomenon and this one's analysis is already a routine activity.

Model generators are software applications taking as input datasets and information regarding model nature, structure and complexity and produce different models from a certain model class.

Linear model generators take as input a dataset containing a number of independent variables and a dependent variable and produce linear models combining influence factors.

Linear model generators with lagged arguments allow the elaboration of constructions which permit the modeling of the cases when a variation in an influence factor has a delayed effect on the dependent variable. The generator produces combinations of both influence factors and lags, identifying best models.

Standard nonlinear model generators use predefined analytical forms for generating models. General nonlinear model generators build automatically analytical expressions containing influence factors.

All generators order their results according to performance indicators.

Model complexity is assessed by measures based on Halstead metrics, taking into account the number of operators and operands. Complexity measures help comparing models and building performance indicators.

The quantitative study of phenomena is encouraged by the volume of historically recorded information. Its purpose is to take decisions leading to continuous improving of performance indicators for subsequent projects.

Modelbases are complex software constructions offering functions for:

- defining, retrieving and updating models;
- modeling applications management;
- estimation and validation of coefficients;
- automated model generation from existing datasets;
- dataset management.

Model generators are modelbase instruments that build model structures from a given class using variables found in the dataset given as input. Model classes group models with the same structure, e.g. linear models, linear models with lagged variables, nonlinear models. For each class a model generator is developed. Each dataset contains data series for the recorded variables. The endogenous variable is specified and the generator builds analytical expressions using influence factors. For each model structure, coefficients are estimated and a performance indicator is computed. The resulting model list is ordered by the performance indicator. The analyst chooses between the best models an appropriate form that later will be used in estimating the studied characteristic.

In the context of model generation, the refinement process is described as follows.

Consider the dataset S and a model generator G . The set of models generated by G to fit the dataset S is L_M . The model set L_M is characterized by the following:

- the number of models is large;
- models have a variety of values for the performance criterion ranging from the worst fitting model to the best fitting model;
- models have a variety of values for the complexity, ranging from the most simple expressions to very complex ones;
- the best fitting models are not compulsory the most complex ones, and also, the simplest models are not the worst in explaining the studied phenomenon.

The refinement process using model generators is described by the following diagram shown in figure 3.

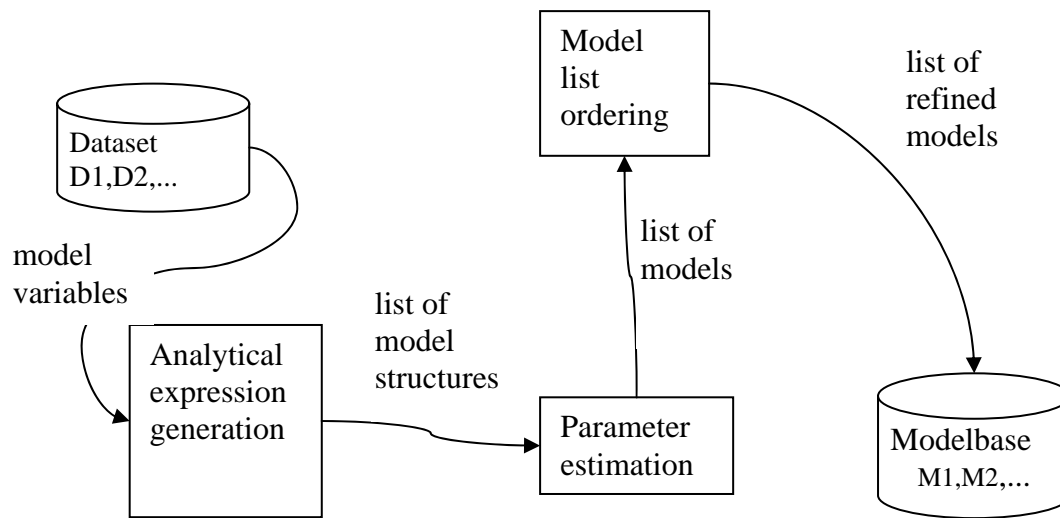


Figure 3. The model refinement process

The refinement process takes the following steps:

- the dataset is built; it contains data series for the dependent variable and independent variables;
- using variable names found in the dataset, model structures are generated; the list of generated model structures is denoted by L_G and contains a large number of models, also depending on the constraints or type of the generation algorithm;
- for each model structure in L_G , coefficients are estimated, along with statistical performance indicators, obtaining the list of estimated models, L_E ;
- for each model found in L_E , an aggregated performance indicator is computed; the L_E list is ordered by this performance indicator and an arbitrary number of models is chosen, forming L_R list, the refined list of models;
- the refined list of models is saved into a modelbase and then used by the human analyst to choose one or more models to be used in estimating the studied phenomenon.

5. Conclusions

Taking into consideration the importance of IT projects the continuance of the study is mandatory:

- Quality characteristics;
- Identifying new indicators;
- Making the data collection and metrics calculation an automatic process;
- Recording reaction to the decision quality to see what are the indicators which need perfecting.

In current time there are instruments used to assist elaboration processes of IT projects. These instruments are meant to ensure the completeness of IT projects.

If the following are built:

- activity lists;

- resource lists;
- job lists.

and for each the number of elements is established from the list, as an instrument is meant to impose:

- the complete description of the elements in each list;
- the correlation of elements between lists;
- the aggregation of data from the generated matrices by the combination of lists.

If a series of relations are imposed regarding:

- the structure of the offer text;
- the expense structure;
- the activity precedence ;
- the rates of financing.

the instruments are meant to:

- verify if the structure is respected;
- generate the parts of structure;
- reduce the number of repeats from one component to the other;
- control the levels of complexity;
- include aggregated levels;
- eliminate abbreviations introducing the elements clearly;
- ensures the qualitative level of the criterions;
- evaluate the level of allocated resources;
- use correct words from a specialty vocabulary.

For a data base in which the records regarding the evolution of a project are saved, the dates must be complete and correct.

If data is taken from several projects the following are calculated:

- differences between the consumption or normed duration;
- The interval belonging frequencies and consumption variation;
- The probability that a variable will follow a certain repartition law.

The complete image over an IT project is obtained only through the implementation of a correctly constructed metric, refined and completely validated.

Bibliography

- Boja, C., Ivan, I. **Metode statistice în analiza software**, Ed. ASE, Bucharest, 2004
- Ivan, I., Boja, C. **Managementul calitatii proiectelor TIC**, Ed. ASE, Bucharest, 2005
- Ivan, I., Nosca, Gh., Popa, M. **Managementul calitatii aplicatiilor informatice**, Ed. ASE, Bucharest, 2006.
- Ivan, I., Popescu, M. **Metrici software**, INFOREC Publishing House, Bucharest, 1999
- Ivan, I., Ungureanu, D. **Project Complexity**, INFOREC Publishing House, Bucharest, 2002
- Ivan, I., Visoiu, A. **Baza de modele economice**, Ed. ASE, Bucharest, 2005
- Jalote, P. **Software Project Management in Practice**, Addison Wesley, 2002
- Toma, C., Ivan, I., Popa, M., Boja, C. **Data Metrics Properties**, Proceedings of International Symposium October 22-23, 2004, Iassy, Romania, pp. 45-56

- Visoiu, A. **Performance Criteria for Software Metrics Model Refinement**, Journal of Applied Quantitative Methods, Vol. 2, No. 1, March 30, 2007
- Visoiu, A., Garais, G. **Nonlinear model structure generator for software metrics estimation**, The 37th International Scientific Symposium of METRA, Bucharest, May, 26th - 27th, 2006, Ministry of National Defence, published on CD
- Visoiu, A., Ivan, I. **Rafinarea metricilor software**, Economistul, supliment "Economie teoretica si aplicativa", no.1947(2973), August 29, 2005

¹ Ion IVAN has graduated the Faculty of Economic Computation and Economic Cybernetics in 1970, he holds a PhD diploma in Economics from 1978 and he had gone through all didactic positions since 1970 when he joined the staff of the Bucharest University of Economics, teaching assistant in 1970, senior lecturer in 1978, assistant professor in 1991 and full professor in 1993. Currently he is full Professor of Economic Informatics within the Department of Economic Informatics at Faculty of Cybernetics, Statistics and Economic Informatics from the University of Economics. He is the author of more than 25 books and over 75 journal articles in the field of software quality management, software metrics and informatics audit. His work focuses on the analysis of quality of software applications. He is currently studying software quality management and audit, project management of IT&C projects. He received numerous diplomas for his research activity achievements. For his entire activity, the National University Research Council granted him in 2005 with the national diploma, Opera Omnia.

He has received multiple grants for research, documentation and exchange of experience at numerous universities from Greece, Ireland, Germany, France, Italy, Sweden, Norway, United States, Holland and Japan.

He is distinguished member of the scientific board for the magazines and journals like:

- Economic Informatics; - Economic Computation and Economic Cybernetics Studies and Research; - Romanian Journal of Statistics

He has participated in the scientific committee of more than 20 Conferences on Informatics and he has coordinated the appearance of 3 proceedings volumes for International Conferences.

From 1994 he is PhD coordinator in the field of Economic Informatics.

He has coordinated as a director more than 15 research projects that have been financed from national and international research programs. He was member in a TEMPUS project as local coordinator and also as contractor in an EPROM project.

² Adrian Visoiu graduated the Bucharest University of Economics, the Faculty of Cybernetics, Statistics and Economic Informatics. He has a master degree in Project Management. He is a PhD student at Doctoral School of Bucharest University of Economics in the field of Economic Informatics.

He is an assistant lecturer in the Economic Informatics Department of the Bucharest University of Economics.

He published 7 articles and he is coauthor of "Baza de modele economice" book.

³ Dragos Palaghita is a 4th year student in the University of Economics, Bucharest, Cybernetics Statistics and Economic Informatics faculty, Economic Informatics section. He is programming in C++ and C# and his main areas of interest are Informatics Security and Software Quality Management.