Software measurers are astonished by the carelessness of IT branch:

Exceeding budget in an IT project is not the law of the nature

If you go to a constructor with your needs, you get a building. Even, if only cleaning the closets would be the thing to do.

The same applies to software projects, clarifies Ms. Carol Dekkers, a Canadian consultant of Quality Plus Technologies. "If you go to a software company, you'll get software. But maybe that is not the best solution to your problems".

Two thirds of software projects are failed. The budget and schedule are exceeded, and the result is not what was intended.

According to Dekkers, the reason is poor planning. Mostly the customer doesn't know, what is really needed. "If you are going to build a house, surely you first acquire the plan, and not just start building."

Dekkers visited Finland and gave a lecture in a seminar of the Finnish Software Metrics Association (FiSMA).

The larger the software project, the greater the odds of failure. This is proven by Standish Group's study. Over a half of the projects carried through by six persons, with a budget of 500,000 dollars and with a six months schedule, are succeeded. Every fourth middle-sized project (25 persons, budget of 1,5 – 3 million dollars), is succeeded. Of over 10 million dollar projects, none is finished in time and/or within budget.

FiSMA's senior advisor, Mr. Pekka Forselius, wonders why e.g. the size of the software is not estimated beforehand. "Software size estimation is easy, but they don't even teach it (*that is not even taught?*) in universities and colleges".

Dekkers and Forselius hate fixed budgets and schedules. And yet, the budget is usually the first thing to be decided, especially within the public sector. The finishing date is more important to companies.

The reason is budgeting. "It is simpler to budget five million dollars for a million dollar project, than to keep the amount open (*unfixed??)* ", states Dekkers.

A fixed budget often leads to a situation where one has to submit to poorer quality and less functionality than required.

Dekkers' solution to these problems is to hire a consultant to analyze what are the customer's actual needs. This is often considered to be too expensive. "Is two or three percent of the budget really too much?"

"But it is easier to spend more money than less money. It's a fact that an amount of money you can't hold in your hand, is more easily spent than a couple of dimes", says Dekkers. "And if you don't know what you want, you will never get it".

Consultant's job is to draw out sketches, define functions and set the level of quality. The most important thing is that requirements don't change all the time in the middle of a project, because then things have to be done all over again.

The definition includes **what** the software does, not **how** it does it. And after this you choose the software supplier.

According to Dekkers, software sizing was first started in Australia. The most important feature of this method is to set a price to every function. Every function has a fixed price. A function is, for example, adding a customer, drafting a contract or sending an invoice.

In planning stage it is determined, how many different functions have to be included in software. Planning beforehand also improves quality, speeds up the project, and increases flexibility. According to Dekkers, most of the errors (*bugs?*) can be traced back to poor definitions and requirements.

"For instance, in the U.S. Army it's been calculated that 40% of the programming work is patching (*error correction/recoding/reprogramming?*). This means coding on Monday, Tuesday and Wednesday. On Thursday and Friday, the work done in the beginning of the week is recoded (*corrected/patched?*)".

The Southern Scope method (*model?*) developed in Australia has given significant results. When software project budgets were earlier exceeded on the average of 84%, the percentage is now on the average of 10.

Measurement is spreading out to Japan, other Asian countries, and to Europe. In Finland, a method (*practice?*) called Northern Scope, is being developed. The Northern Scope differs a little from Southern Scope.

In the USA the situation is problematical. Software standardization is completely in a neglected condition, and this makes quality measurement difficult.

Software measurement is advanced by International Software Benchmarking Standards Group (ISBSG). In Finland, the work is conducted by FiSMA.