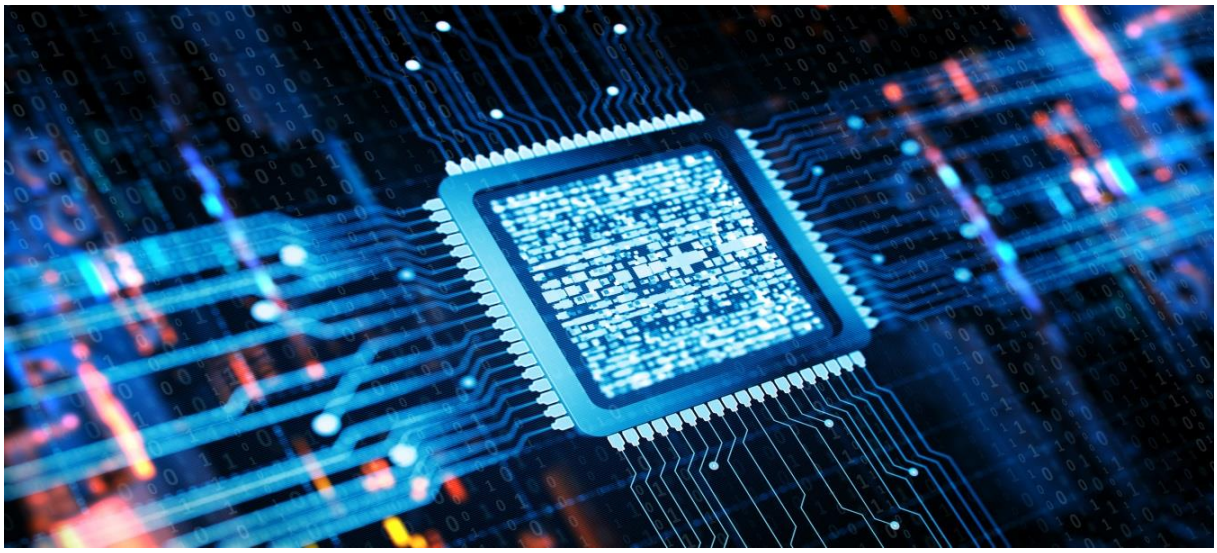
Christian von Mezynski

Evaluating and negotiating software costs – well worth the effort!

THE POLARIXPARTNER SOFTWARE-COSTING MODEL



Electrical components, which include software elements are on the rise. Hardly any area of industry can afford to ignore digital transformation over the long term. Nevertheless, in many organisations, cost engineering processes are not yet up to the challenge. While steady progress is being made in establishing software skills within teams, the shift is by no means complete, and many structures are generally still geared towards cost analysis of production parts, which of course is completely different to software development cost.

At the centre of this white paper is the model developed by POLARIXPARTNER for evaluating software costs. This approach is based on agile software development processes. It examines the resources required in great detail, while at the same time ensuring the efficiency necessary to carry out the evaluation within an acceptable time: the insights gained are a solid base for subsequent price negotiations. In the following sections, the software-costing model developed by POLARIXPARTNER is outlined in more detail.

THE STATUS QUO OF COST EVALUATION

Forming groups for the purposes of cost evaluation is an increasingly popular approach. Many car producers have already established entire departments, and other industries are increasingly following suit. Hardly any area remains unaffected: costs for parts made of synthetics, metal, glass, textiles and many other raw materials are evaluated; production processes are analysed in detail and the appropriate cycle times and hourly rates for machines are calculated exactly, right down to the euro cent; for electronics components, SMD modules with a size of up to 0.1mm are counted and cost-evaluated.

Where software evaluation is concerned, a different picture is emerging because it is subject to different rules. Firstly, the activity in focus is pure development. Secondly, the software-development process differs from other development processes: there are no overhead costs for processing in the original sense and fix costs are different; overhead targets are defined differently too. Many teams lack the underlying knowledge to comprehend the necessary development steps for software production. Currently, all these factors taken together are a barrier to negotiating with software suppliers as equals.

THE DIFFERENCES

In order to evaluate required software-development resources, some basics need first to be explained. Software development occurs through processes, which are barely comparable or wholly different to familiar ways of proceeding in the areas of electronics hardware development or mechanical design. The following two sections describe the individual software development processes and how overheads are structured.

| > SOFTWARE DEVELOPMENT PROCESSES

Hardly any initiative within the software world has changed development processes as much as agile development methodology. For a long time, programs were developed fully before confirming that they corresponded to customer requirements – with the result that whole months of developer time were sometimes wasted because the delivered software did not align with client expectations. The agile approach was born out of experience with this waterfall principle. Here, the development work is broken down into small packages which are implemented and tested in their entirety. The development of these packages extends over two to three weeks. If the result of the software programming departs from customer needs, the course can be corrected in a timely manner within the next development package, or "sprint".

The evaluation of software costs should, in principle, be based on the assumption that development is being conducted according to agile methodology. We differentiate between cases where the scope is merely broken down for programming and testing in small packages (e.g. iterative V model), and those where a comprehensive agile development methodology is employed (e.g. SCRUM). The following development stages constitute the foundations of the POLARIXPARTNER software costing model:

Figure 1: Development stages

Activities, which play a supporting role to the actual development are called development-accompanying process stages. The following are defined within the POLARIXPARTNER software-costing model:

Figure 2: Development-accompanying process stages

Every process stage includes a detailed specification with clearly defined activities. It is not only the differences between various development methods which are taken account of, but also aspects such as functional safety, re-use factors, SPICE requirements, diagnostics, real-time behaviour and many more.

| > OVERHEADS STRUCTURE

Two elements have remained unchanged compared to the manufacturing industry: profit and sales/administration. Here, the same calculation principles apply: it is only the target values which differ – and have therefore been thoroughly revised. As software is not a tangible product, the cost structure differs noticeably:

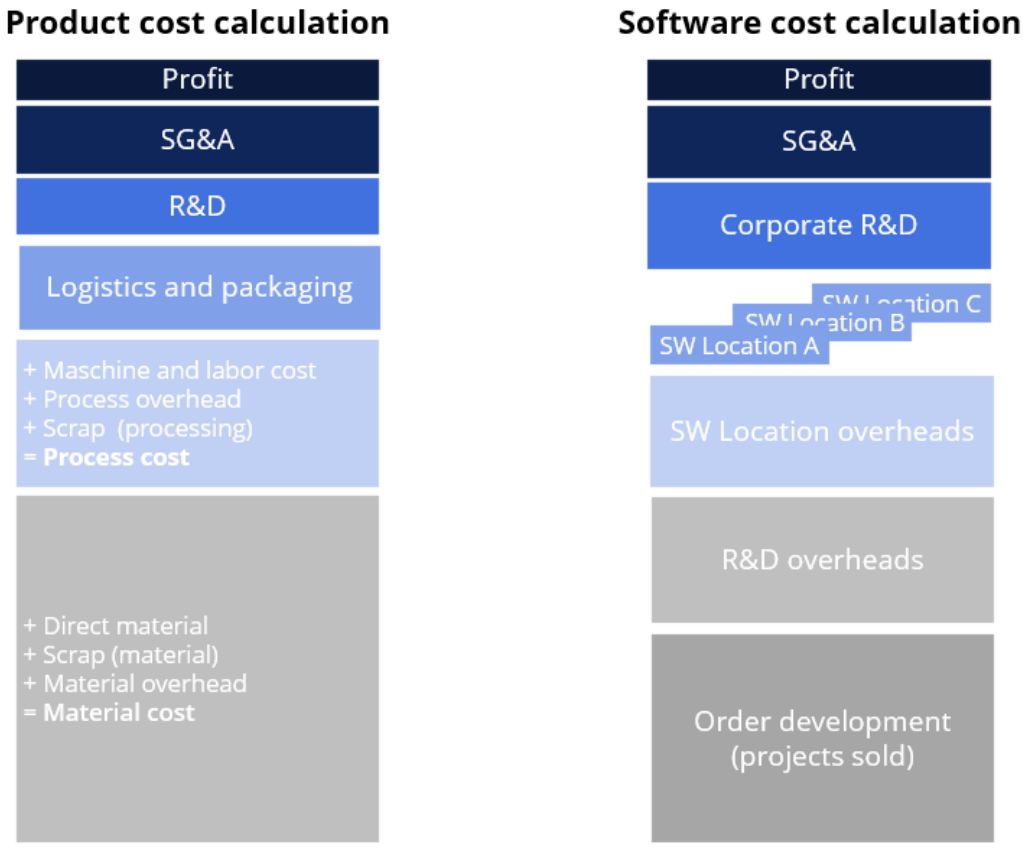


Figure 3: Structure of software costs

The various overhead costs are further detailed in the software-costing model. With the help of this information, a calculation can be produced that is an exact fit for the business structure in question. The development work itself is termed “order development”: it generates the added value that is later invoiced. This is where the development steps (detailed above as software-development processes) take place. In terms of its applicability, the POLARIXPARTNER software-costing model can be used in development both of embedded controlled applications (vehicle controllers, e.g. in cars, trucks, aeroplanes, coffee machines, etc.) and of mobile applications.

EVALUATING SOFTWARE COSTS FOR NEGOTIATIONS

Negotiating software pricing requires a fact-oriented discussion in which the negotiators face each other at the same level, and in order to achieve this, the buyer needs to acquire substantial expertise regarding the specific implementation of the software. The POLARIXPARTNER software-costing model outlines an approach borrowed from agile software development, which enables an

efficient evaluation of the relevant development stages without a time-consuming, detailed analysis of each individual function. Our method for evaluation is divided into six steps:

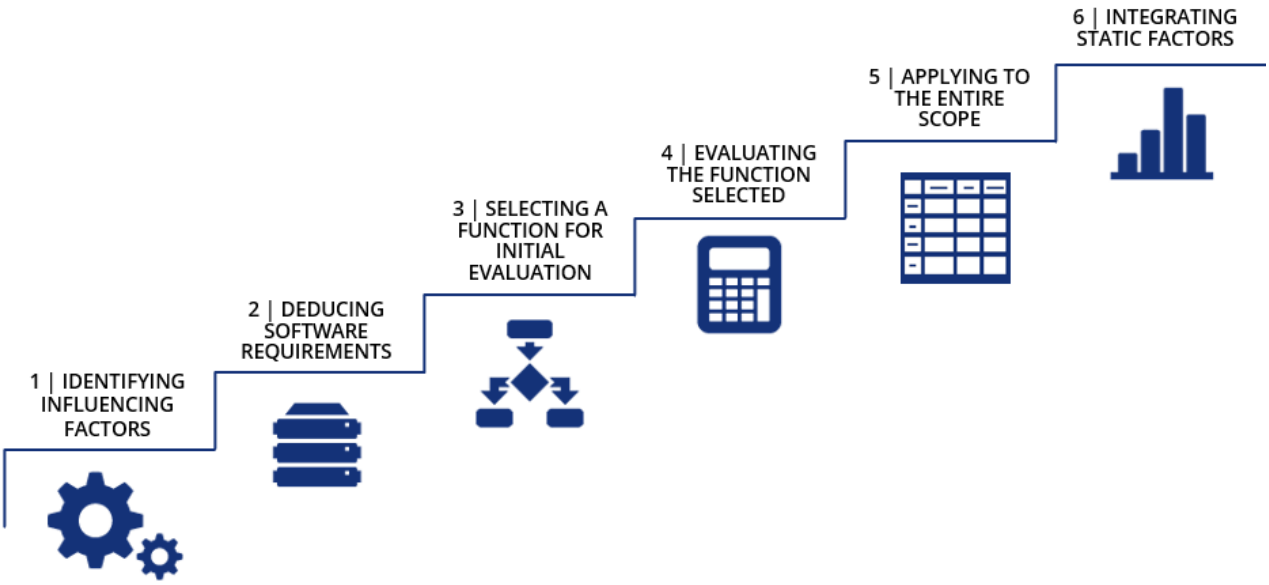


Figure 4: Six steps of software evaluation

The procedure illustrated provides a comprehensive overview of the basic structures of the software application at hand: software architectures are exposed and component specifications allow a deep insight into the necessary programming resources; all necessary process steps are analysed and evaluated in terms of required resources. When all six steps of the software analysis have been conducted, a comprehensive picture of the actual development resources results.

To support this structured documentation of the resources required, a calculation tool precisely tailored to the software requirements is available. It provides standard values for overheads within various business categories, thus making it possible to differentiate between tier-1 businesses, software service providers and start-ups; moreover, the overhead rates can be adjusted to the individual requirements of a software provider. To increase the accuracy of hourly rates, information specific to the countries and regions that are relevant to software programming is provided.Alongside documentation of the resources required for software development, a complete picture emerges which details the costs actually incurred during implementation.

COST CALCULATION

Time and time again, the results of evaluations using the POLARIXPARTNER software-costing model confirm the initial impression that a re-think regarding software-pricing negotiations is necessary: the model often identifies potential reductions of more than 40 % compared to the offer price. By way of illustration, the following real-world example originates from the evaluation of an embedded-controller application within the automotive industry.

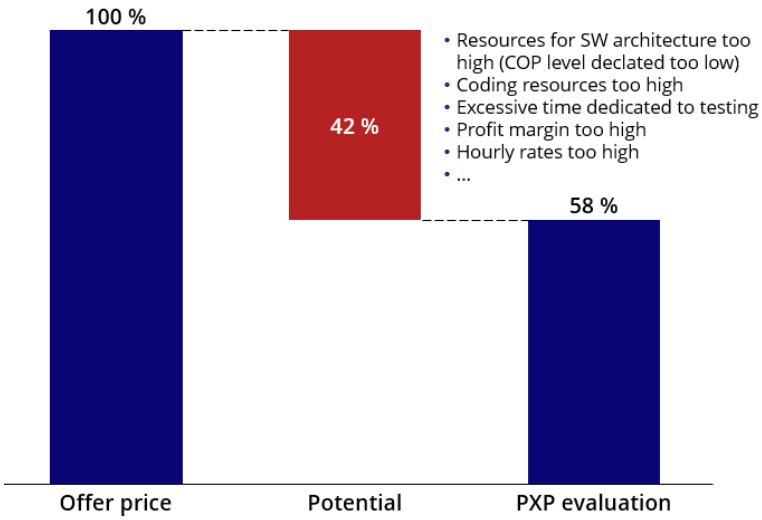


Figure 5: Real-life example: embedded-controller application

The evaluation identified a broad range of areas whose costs were too high. This could be seen in developer services, where too many resources were allocated, as well as in overheads, e.g. the profit margins declared. This real-world example shows that engaging in a technical discussion of software pricing without accumulating software evaluation knowledge can lead to a disadvantageous result.

SYNERGIES

By accumulating knowledge of software implementation, further synergies arise when the projects are carried out:

1. **Comprehensive requirement specifications**

   In many cases, requirement specifications leave lots of room for speculation. When requirements are unclear, software providers can be obliged to make scope decisions. This can lead to expensive revisions if the client originally wanted something else. The software cost-evaluation exposes gaps in the requirement specification, which can be corrected to avoid wasting expensive development resources.

2. **Time planning**

   Software cost-evaluation provides a detailed inventory of resources required for the various project stages. By comparing it with the software provider's time plan, it is possible to check whether certain activities are not accounted for or have been forgotten. This comparison, and any corrections if they diverge from one another, enables an improvement in the quality and completeness of the time plans.

3. **Software quality**

   If the software evaluation lists test activities that the software provider does not carry out, this indicates a quality risk that should be openly discussed. In this case, it is possible to identify at an early stage whether the software provider needs to make improvements to their quality-control processes.

With the POLARIXPARTNER software-costing model, it is possible to gain a thorough understanding of the software product and to deduce the correct actions from this. This makes the risks associated with any software project more manageable.

CONCLUSION

The POLARIXPARTNER software-costing model makes it possible to evaluate software resource requirements quickly and thoroughly. The knowledge acquired through this process enables software pricing to be negotiated at eye-level. Even businesses traditionally rooted in mechanical engineering can understand software resourcing and influence it to their own advantage.

The knowledge of software implementation enables not only fair software negotiations but also an estimation of project risks. Gaps within specification catalogues are exposed, time plans are put under the microscope and the quality of the software services becomes tangible. All of these topics cause costs that were rarely evaluated prior to awarding software contracts yet.

Especially in the automotive industry, but also in many other industries, the prevalence of software is continually increasing due to advancing digital transformation. Challenges such as electrification and autonomous driving require software functions capable of carrying out extremely complex calculations in very short times. Moreover, many businesses see themselves forced to source mobile applications. The evaluation of the necessary software applications is gaining more and more importance. To prevent costs running out of control, we accompany our customers on their journey towards a transparent cost evaluation for software applications.

THE AUTHOR & YOUR EXPERT CONTACT AT POLARIXPARTNER

- Experience in the areas of electronics hardware/software development specialising in: verification/validation, product specification, project management and product-cost optimisation
- Team member in various roles under SCRUM
- Optimisation of development processes affecting agile software development
- Expert and HR leadership of developer teams and project leads
- Market analysis of developer hourly rates in various countries worldwide
- Cost evaluation of all types of software

Christian von Mezynski

POLARIXPARTNER GmbH
Graf-Siegfried-Str. 32, 54439 Saarburg, Deutschland
www.polarixpartner.com

Tel. +49 6581 8290-266
Mobile +49 151 52742526
E-mail Christian.Mezynski@polarixpartner.com

ABOUT POLARIXPARTNER

MANAGEMENT. CONSULTANCY. IMPLEMENTATION. POLARIXPARTNER is the management consultancy for the manufacturing industry. As industry insiders with many years of experience, we guide you on your way to success. Our approach is holistic, our philosophy focussed on implementation: we analyse and strategically evaluate your core processes while remaining active on your shop floor of your company. We deliver improvements along the entire value chain: from the process of change in the product due to technological change, to the processes and systems responsible for adding value, to all accompanying support functions. Together with our customers, we develop efficient and forward-looking concepts - fast, result-, goal- and implementation-oriented. **THINKING AHEAD. OPTIMISING. IMPLEMENTING.**