Combining Product Line Engineering with Options Thinking

Birgit Geppert, Frank Roessler
Avaya Labs- Software Technology Research
233 Mt. Airy Road, Basking Ridge, NJ 07920, USA
{bgeppert, roessler}@research.avayalabs.com

Abstract

Developing a product family in an uncertain environment where future family members cannot be reliably predicted generally implies high risk for a product line approach. As managers usually tend to favour short term and low risk projects, such a product family is not likely to be engineered as a product line. However, we can engineer different evolutionary paths of a product that are uncertain but have high potential of revenue growth as a product line also. One would start with developing common aspects of possible evolutionary paths and move on with variabilities while uncertainty is resolved. Because of time-to-market advantages the inherent flexibility of the resulting architecture can have great value for an organization and therefore justify a product line approach. In this context variabilities are considered as options that give you the right but not the obligation to evolve the product in one of several possible directions. This paper introduces the basic idea of *Strategic Product Line Engineering* and gives an outlook on how option-pricing theory might be applied to estimate the value of strategic product lines.

Keywords. Software Product Lines, Strategic Software Design, Real Options

1. Strategic Product Line Engineering

The concept of *software families* has been well-known for decades [8]. However, developing a family of programs is not necessarily equivalent to developing it *as* a family, i.e., systematically exploiting the family members' commonalities and controlling their variabilities. Such exploitation results in a *software product line*, which usually includes an infrastructure that enables rapid development of family members.

A common problem of product line engineering as documented in [2], [10], is that the infrastructure can require a substantial initial investment. This investment is returned through the accumulated savings we obtain by deriving the family members from the common infrastructure. Figure 1 illustrates the underlying economic model: after a certain number of family members have been produced, the investment is repaid. From this point

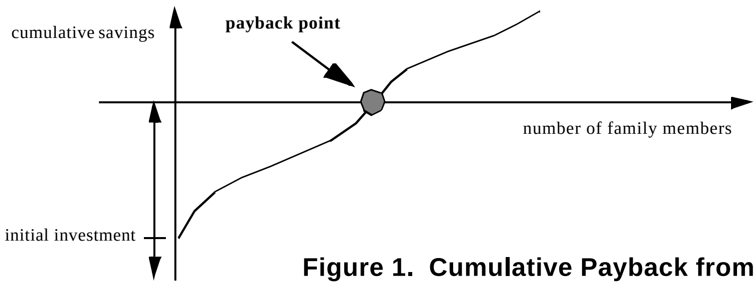

Figure 1. Cumulative Payback from Product Line Engineering

on, we expect net return on the investment in the product line. Note that the shape of the curve depends on the specific project. In particular, the amount of initial investment varies and can be influenced by different adoption strategies (in specific cases initial investment may even be zero). In this paper, we are considering projects where some initial investment is deemed to be necessary or advantageous and seek for possibilities to justify it.

The economic model of Figure 1 fits very well when we consider product families with several family members produced at the same time, possibly in regular intervals, as shown in Figure 2. In such a situation, we can reliably predict family members and can expect the payback point in the near future. Therefore an initial investment can be justified without taking too much risk. An example is a cellular phone vendor, who produces about 10 new phones every year. Old phone models are not maintained but replaced with more advanced models.

However, the situation is different if we consider an evolutionary product that is maintained and evolved over time resulting in several releases instead of new models. We can still consider the set of releases as a family, but can we justify developing them as a product line? An example for such an evolutionary product would be a PBX.

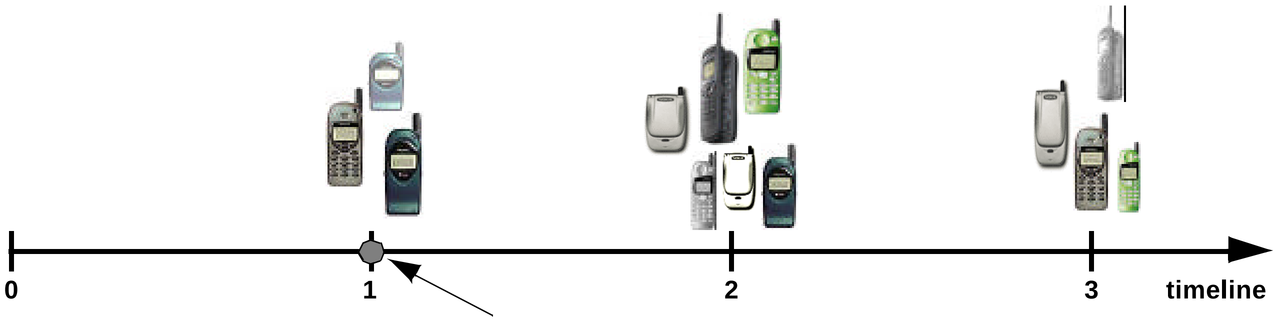

Figure 2. Product Family in Space: Cell Phones

However, because product line engineering is currently not mature enough to be applied to such complex systems at a whole, we apply it only to selected subsystems and therefore also only get part of the potential savings. Furthermore, call processing software is too complex to allow for many variants at the same time[1]. Instead, the existing product is gradually evolved as illustrated in Figure 3. Family members are spread out over a long time, and therefore a potential payback point is located further out in the future. This increases the risk when deciding on a product line approach, because we cannot well predict what family members would be needed. Since managers tend to favour short term and low risk projects, a product family in time is generally a less likely candidate for a product line.

But does this mean that product line technology should not be applied to evolutionary products at all? Definitely not! Instead of engineering the family in time as a product line we should think of possible evolutionary paths as a product line. Figure 4 illustrates this for the PBX example: assume we are at time 0. At that time, we can already reliably predict that there will be four possible variations for release 2 of our product. Which variation will eventually be most successful depends on future events that we currently cannot control or reliably predict (such as customer reaction to release 1 or announcements of competitive products). Nevertheless, it might be beneficial to provide for sufficient flexibility in the software architecture of release 1 to quickly move to one of those different variants when release 2 is due (in the example, we have chosen to only support three of the predicted variants). These variations can be provided through common product line technology, but we prefer calling this approach *Strategic Product Line Engineering*. We consider the predicted variants for future releases as a (alternate) family and create an adequate PL infrastructure for it. It is important to understand that variabilities in this context must be options that give you the right - but not the obligation - to evolve the product in one of several possible directions[2].

One might think that this is actually a waste of resources because not all considered family members are going to be developed and not all incorporated options are going to be exercised. However, we argue that having a portfolio of options in the architecture - even without guarantees that they all will eventually be exercised - can create *immediate* value and therefore justifies developing the alternate family as a product line.

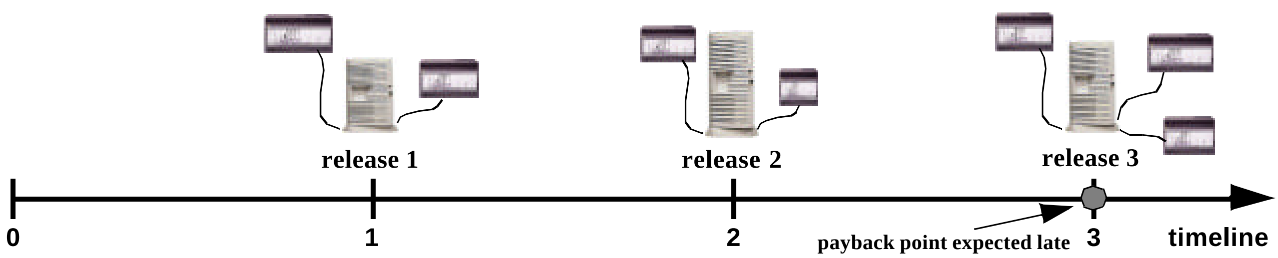

Figure 3. Product Family in Time: PBX

[1] Let us assume that there is no customization such as different line sizes, different billing systems, or country-specific needs.
[2] In particular, this means that costs for “encapsulating” the variability in the design should be low compared to the costs for implementing a variant.

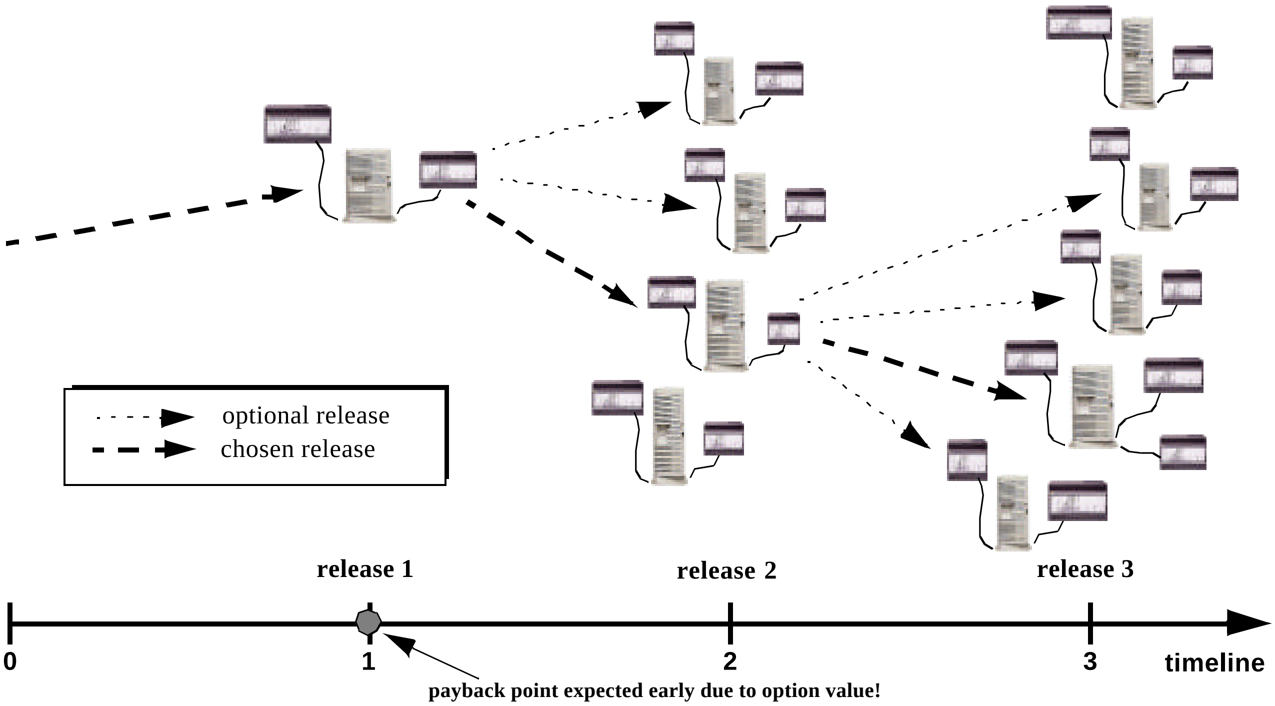

Figure 4. Strategic Product Family: Option Space for PBX

2. Value of Strategic Product Lines

The economic model of Figure 1 for estimating the value of a product line is based on a common method for valuing investments called *discounted-cash-flow* (DCF) *analysis*. Like any other valuation technique, it is a function of the three fundamental factors: cash, timing, and risk [6]. DCF analysis identifies all cash flows related to the investment, such as expected revenues from the different family members, costs for conducting the commonality analysis, or costs for building code generators. This yields a series of cash flows stretching into the future. Those cash flows are adjusted for time and risk and then summed up to yield the value of the investment. Only if the value is greater than zero, do we recommend proceeding with the project.

The main problem with DCF calculations is that they are only valid when valuing an ongoing business or an immediate investment. DCF analysis can estimate how much expected future cash flows are worth once an investment is made. Hence, as an analysis tool for managers, DCF can only support go or no-go decisions. It does not account for the ability of managers to react to new circumstances - for instance, to spend a little up front, see how things develop, and then either cancel or go full speed. However, strategic product line engineering is based on exactly this kind of flexibility in the product and project structure. In order to estimate the value of strategic product lines, we therefore need a valuation approach that considers flexibility in decision-making. In the following we briefly illustrate such an approach.

In [3], Black and Scholes proposed an analytic model for determining the fair market value of a call option, which resulted in the so-called *option-pricing theory*. Today, this is one of the most widely accepted financial models[1].

The idea of options was certainly not new. When used in the financial world, options are generally defined as a *contract between two parties in which one party has the right but not the obligation to do something, usually to buy or sell some underlying asset at a specified price on or before some future date*. In particular, call options are contracts that give the option holder the right to buy something. A call option on a share of stock gives you the right - but not the obligation - to buy that share for, say, $100 by the end of this year. If the share is currently

[1] Myron Scholes together with Robert Merton were rewarded with the Nobel Prize in economics in 1997.

worth more than $100 the option certainly is valuable. But even if it currently sells for less than $100, it might still be valuable, because stock prices can rise before the option expires. Thus, having rights without obligations has financial value, a value that option-pricing theory can calculate very accurately.

There are only a few parameters that influence the value of an option. The most important ones are the volatility of stock price and the time to expiration. The expected rate of return of the underlying asset is not one of the variables in any model for option valuation. This means that a prediction of the future price of the underlying asset is not necessary to price an option. Thus, while any two investors may strongly disagree on the expected rate of return on a stock, they will always agree on a fair price for the option, as long as they agree on volatility and the risk-free rate. Since 1973, the original Black-Scholes Option-Pricing Model has been expanded in many ways and several of its assumptions have been relaxed, resulting in amazingly accurate valuation models for stock options.

To apply an option-pricing model to (real) options that are embedded in a strategic product line we must find a mapping from project characteristics onto the variables of the option model [7]. Let us assume an option allows us to replace quickly a protocol standard used in our PBX. It is only necessary to exchange a module that implements the new protocol. This investment opportunity is like a call option, because the software organization has the right - but not an obligation - to develop the new module. If we could construct a call option sufficiently similar to the investment opportunity, the option price would tell us something about the value of the investment [6]. For instance, we can easily map the project characteristics onto the five variables of the Black-Scholes Model: spending resources for developing the module is analogous to exercising the call option, i.e., the development costs correspond to the option's exercise price. The present value of the module, including module development costs and expected savings from product line engineering, correspond to the stock price. The release date corresponds to the expiration date of the call option. The uncertainty about the future revenues related to the module corresponds to the volatility of the stock price.

The option we construct this way is certainly not a perfect substitute for the real option (i.e., the module investment opportunity), because not all assumptions of the Black-Scholes model will be met by our project. However, even with the basic Black-Scholes Model we expect to get better insights into our project than with a simple DCF analysis. However, there remain difficult questions about a reasonable estimation of certain variables of the Black-Scholes Model such as volatility, and whether certain assumptions of option models can be met in the non-financial world. The literature in the field of real options promises answers to these questions ([1], [4], [5], [9]) that must be further investigated in the context of product line engineering.

3. Summary

In this paper, we introduced the basic idea of *Strategic Product Line Engineering*, a method that engineers possible evolutionary paths of a product as a product line. In this context, variabilities are considered options that give you the right but not the obligation to evolve the evolutionary product in one of several possible directions. We also illustrated that real option theory has the potential to support strategic product line engineering by helping managers to quantify the value of active management and strategic interaction embedded in various product line investment opportunities.

References

[1] C. Y. Baldwin and K. B. Clark, *Design Rules - The Power of Modularity*, MIT Press, 2000

[2] J. Bayer et al., *PuLSE: A methodology to develop software product lines*, Symposium on Software Reusability, 1999

[3] F. Black and M. Scholes, *The pricing of options and corporate liabilities*, Journal of Political economy, 1973

[4] T. Copeland and V. Antikarov, *Real Options: A Practitioner's Guide*, Texere, 2001

[5] J. Favaro, K. Favaro, and P. Favaro, *Value based software reuse investment*, Annals of Software Engineering 5, 1998

[6] T.A.Luehrman, *What's it worth? A general manager's guide to valuation*, Harvard Business Review, May/Jun 1997

[7] T.A.Luehrman, *Investment Opportunities as Real Options:Getting Started on the Numbers*, Harvard Business Review, Jul/Aug 1998

[8] D. L. Parnas, *On the Design and Development of Program Families*, IEEE Transactions on Software Engineering, No. 1, March 1976

[9] K.J. Sullivan et al., *Software design as an investment activity: a real options perspective*, in *Real Options and Business Strategy: Applications to Decision Making*, L. Trigeorgis (Editor), Risk Books, 1999

[10] D. Weiss and R. Lai, *Software Product-Line Engineering - A Family-Based Software Development Process*, Addison Wesley, 1999