

Optimization of SME Financial Engineering assisted by a Software Agent

by Dr. Mathias Künlen

Liquidity planning and monthly target-performance comparisons are common practice in small and medium-sized enterprises. Liquidity planning, as the name implies, represents a plan but does not guarantee performance of the values provided in the planning. The sales and costs numbers more or less result from past experiences and future expectations, hopes, and aspirations. This is where the danger lies against the background of Basel2/Rating: the lending banks pin the entrepreneur down to his planning. Planning that does not keep up with reality will lead to a downgrade in rating and thus to a rise in the cost of current account overdraft facilities. Long-term liquidity planning systems have no operational effect in terms of reorganizing payment flows and optimizing liquidity when liquidity squeezes arise.

The author uses a practical example to show how an SME manages to not only plan short and medium-term liquidity developments by implementing a software agent in addition to long-term liquidity planning, but to forecast them with a high degree of accuracy on a daily basis. The software agent is also able to manage corporate finances according to optimizing criteria. This leads to much greater security in the company's financial engineering, increased profit margins, and improved ratings. The software agent operates fully automatically in the background of the financial accounting system used.

Many SMEs have a financial engineering system which is largely fed from the current account. However, a high credit line costs money with commitment interest having to be paid in addition to current account rates. It is important for the company to convert static liquidity planning into dynamic liquidity management in order to reduce the current account borrowing need and to improve the company's liquidity situation without outside help. This can be achieved with a software agent, working on the basis of artificial intelligence, learning how to adapt on an automatic regulatory basis and managing processes independently.

Case study:

In the past year, the company Nemo Ltd (name changed), operating in the publishing business (production and sale of print products), had a sales volume of 53 million € with a profit margin of 3% and a current account overdraft limit of 3 million €. For the current year, at an equity ratio of 32%, the total sales volume predicted was 6% below that of the previous year. The general market situation was estimated to be difficult, particularly as foreign competitors had taken on a number of large orders at more favorable conditions.

The company management complained that the long-term liquidity planning process was very time-consuming, while financial management processes could not be optimized. The planning tool showed expectations for the future but did not offer any realization strategies for actually achieving these goals. In that sense, planning was static and required comprehensive reworking if the general conditions changed, e.g. due to sales campaigns.

The monthly target-performance evaluation of the current liquidity planning showed that the forecast developments were congruent with real events only to a very limited extent. It was frequently the case that, due to deteriorating payment behavior for example, liquidity inflows did not occur at the rate predicted. It was at this point that the liquidity planning weaknesses were also revealed: debit-side payment targets were only ever applied to debit-side open items in the

form of standardized values (30 and 60 days payment period net). This resulted in severe systemic errors, particularly since the debtors' individual payment behavior was not taken into account.

Even though the current liquidity planning appeared to be future-oriented, it was in fact no more than a retrospective process. Target-performance evaluations dealt with the past without the company management having the chance to intervene operationally or to carry out any form of optimization.

Neither did the model used to date provide any opportunities for identifying hidden liquidity risks easily and quickly. In fact here the evaluation of the financial accounting data required considerable effort, with no guarantee of the right solution.

The following functionalities constitute the main features of software agents:

1. Debit-side payment behavior evaluation
2. Learning from periodicity
3. Liquidity forecasts on a daily basis
4. Identification of latent risks in customer structures
5. Early detection of liquidity squeezes and active countermeasures
6. Rating-compliant disposition management
7. Consistent realization of optimization potential

Software agents:

Software agents are computer programs capable of a certain degree of independent behavior.

Artificial Intelligence research defines a software agent as software with the following properties:

1. autonomous – the program operates largely independently of user interventions
2. learning – the program learns from decisions and/or observations made previously
1. proactive – the program triggers actions on its own initiative
2. reactive – the program reacts to changes in the environment
3. social – the program communicates with other software agents

Software agents are characterized by knowledge, the ability to learn, conclusions, and the ability to change their behavior. From among the different types of software agents (observing software agent, reactive software agent), so-called cognitive agents represent the most sophisticated type of software agents. They maintain a model of their environment in their own data structure, and they learn and adjust. The decisive factor here is their ability to work in interactive models, to separate important from unimportant interaction impulses, and to independently draw conclusions for forecast developments. This enables action planning and subsequently also target-oriented action.

This means that we are experiencing a **paradigm shift**: the user no longer has to take action on his own initiative every day; instead software agents do the planning and steering as highly independent and fully automatic systemic cognitive amplifiers. Unlike conventional applications, the resulting process strategies are not the outcome of given “reproducing” processes, but are developed and realized dynamically by software agents with independent cognitive competence. The software agents therefore typically avoid the errors in reasoning and perception that human users tend to make.

1. Debit-side payment behavior evaluation

The company is able to have the customers' payment behavior monitored fully automatically in the background from the moment the software agent is introduced. To do this the system uses bank entry booking to record the extent to which the respective customers keep to the agreed term of payment, whether they deduct a discount, whether this is legitimate, whether they deduct a discount even if the terms have been exceeded, etc. The important factor is the proactive nature of the system meaning that the user does not have to take any action in order to acquire knowledge of current payment behavior. While the specific debtor analysis deals with each customer individually, the anonymous debtor analysis uses further characteristics for analyses and evaluations covering all customers. The important criteria here are seasonal sales profiles, dependencies of payment receipt probabilities, interactions with the billing date, amount validations, and many more. It is the creeping changes in particular, detected only to a limited extent within conventional evaluation procedures, that are being identified and reported by the agent system.

Example: In the company analyzed there is sufficient information on 50% of the approximately 2000 debtors in the customer list in order to carry out a specific payment behavior evaluation. 40% of the other half can be evaluated through anonymous cluster procedures, with sufficient accuracy for including the data in the liquidity forecast calculations.

2. Detecting and learning from periodicity

Continuous monitoring of credit-side debiting, direct debiting or standing orders may appear routine but it causes severe problems in the company's daily life. It is easy for humans to detect something when it is present. Detecting something if it is not present, but should be is more difficult, however. This applies in particular to overdrawn payment transactions which should have been debited in the meantime. Based on its periodicity knowledge developed automatically, the software agent creates expectation scenarios on its own that are compared with real developments. This enables the company to quickly detect whether debit entries have not been collected, to take these into account several times if required, and to build up appropriate reserves in order not to get caught in a liquidity squeeze when the entire amount is debited in one sum.

Example: In this way the company can elegantly buffer the repeated omission of a direct debiting procedure for leased copying machines. The software agent realizes that the debits corresponding to the periodicity to be expected have apparently not been collected for several months for unknown reasons and automatically creates appropriate reserves within the cash management process. The amount of 80,000 € accumulated in the meantime continues to be reserved for the expected debit transaction and is treated as a separate transaction in the liquidity management.

3. Liquidity forecasts on a daily basis

Thanks to the accurate debit-side payment behavior evaluations and the profiles for periodically recurrent payment transactions learned by the agent system, the company is now able to calculate liquidity forecasts on a daily basis. To do this, the software agent takes the open items (on the debit/credit side) and budget values from the financial accounting/ERP system, reviews them using the acquired knowledge on payment behavior, periodicity, interaction profiles, etc, and corrects the dates for debit-side payment receipts on its own. Compared to static liquidity planning, this creates an altogether more accurate image which corresponds to reality.

Example: Using the daily forecast calculations, the software agent is able to identify a liquidity squeeze in 2 weeks time, which will probably last for three days amounting to approx. 20,000 € and will then resolve itself in the course of further liquidity development. The short-term nature of this deficit would not be identified in this manner with a conventional planning approach. At the same time, the software agent proposes delaying the payment of 3 defined liabilities by a total of 5 days as part of a lagging strategy in order to avoid the impending liquidity squeeze. The early timing of the message and automatic development of the appropriate solution strategy provides a high level of entrepreneurial security in financial management, and additional outside funds are no longer needed.

4. Identification of latent risks within customer structures

Through continuous evaluation of customer payment profiles the agent system identifies impending risks in customer structures, something the company was not capable of doing in this manner previously.

Example: The company achieves a sales increase of 5% compared to the original planning. After evaluating the business numbers it becomes apparent that this sales increase is due to one specific customer in particular who places orders at regular intervals and receives the corresponding deliveries. What is initially judged to be a promising development in terms of increased sales turns out to be a risk when looked at in more detail. The customer keeps building up an inventory of receivables (increase of 25% with progression), delays the agreed terms of payment more and more (progression from payment on time to dunning level 1) and changes his bank accounts. All this indicates that the customer is increasing the order volume only because he has already deterred other suppliers and can no longer obtain credit terms, as well as the fact that his banks have introduced restrictive measures in the meantime.

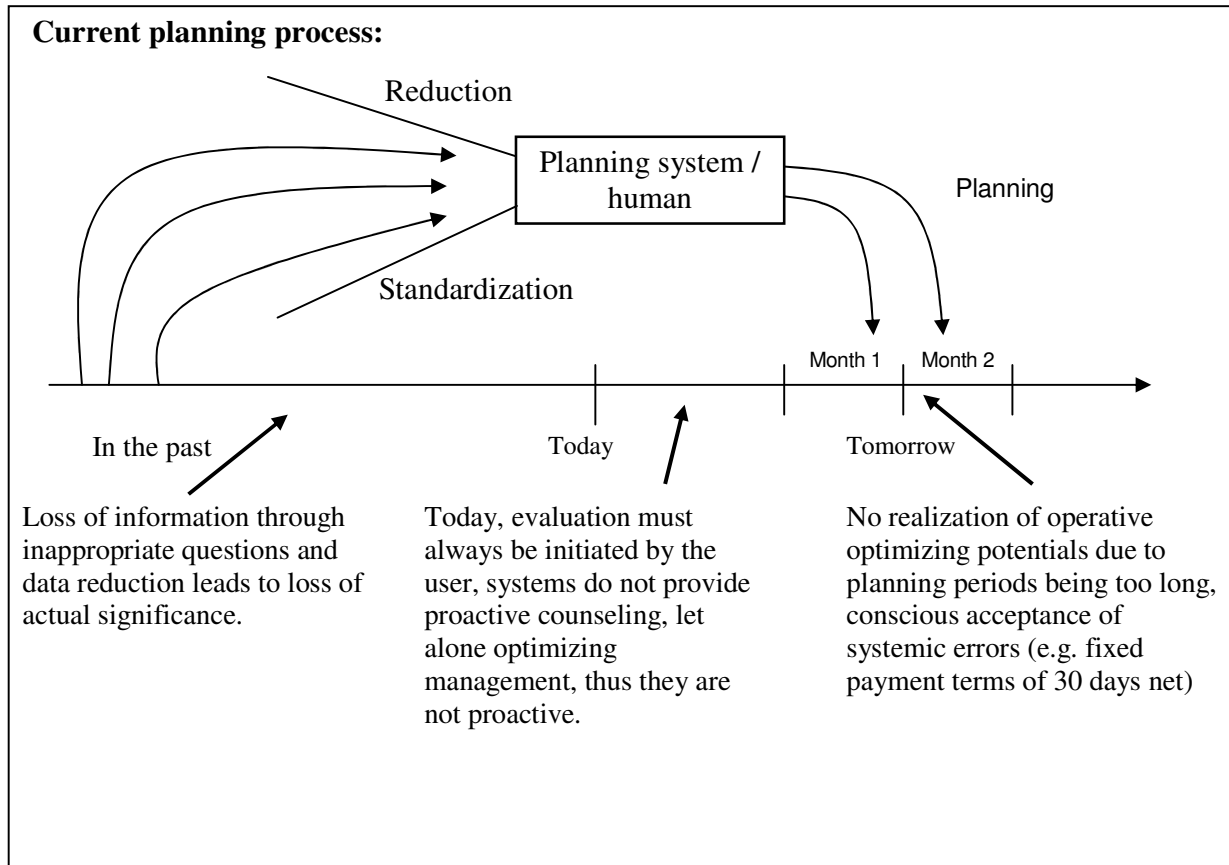
This situation makes it very clear: The controller is misled by his perception of the increasing sales and it is only after closer examination of the context that he recognizes the information which is really relevant, in this case the impending risk for the company due to a customer who can be expected to become insolvent soon. What seems to be a paradox at first glance turns out to be a logical constellation in the end. This leads to serious problems in practice, particularly as the controller usually realizes this type of situation much too late.

5. Early detection of liquidity squeezes and active countermeasures

Short-term liquidity squeezes lasting only a few days are not registered by conventional, static liquidity planning, since the set planning periods are too long. This situation does have unpleasant consequences, however, because it will already be too late when the company management negotiates with the bank for a short-term increase of the credit line. This could lead to a downgrading in the rating and will then involve higher interest cost on the current account.

Example: Based on daily forecast calculations, the software agent can now also identify short-term liquidity squeezes at an early stage and automatically report them to the user. This is important especially if the impending liquidity squeeze can be avoided without outside help by skillfully reorganizing payment flows. The software agent informs management of a liquidity squeeze coming up in 12 days, threatening the company with an amount of 35,000 € for 4 days. After that the squeeze will resolve itself in view of the debit-side payment receipts to be expected amounting to 42,000 €. The software agent describes possible solution strategies, e.g. shifting 3 liabilities by 2-3 days each, without the need for a further increase in the credit line.

The same principle applies in the reverse: If an unexpected liability occurs which has not been considered in the planning so far, the user will not have to content himself with conventional “what-if-scenarios”, but will receive concrete proposals for installment or lagging strategies. These can be implemented operationally by a mouse click.



Example: with cash currently available amounting to 0.55 million € and a current daily liquidity amounting to 0.4 million €, the software agent forecasts an imputed liquidity of 0.35 million € in 10 days. To replace an inoperable printing machine the company simulates a new liability amounting to 400,000 €. The software agent proposes an installment strategy of 3 installments of 0.35 million € in 10 days, 0.03 million € in 22 days and 0.02 million € in 51 days. The liquidity coverage is therefore not exceeded and additional borrowings are avoided at the same time.

6. Rating-compliant disposition management

Rating-compliant disposition management aims at performing all disposition activities in consistent compliance with the liquidity developments calculated by the software agent and according to the banks’ rating guidelines for accounting. With no liquidity squeeze impending, disposition can be managed in the regular manner. All payments can be carried out according to the defined terms and conditions of payment.

However, if the software agent forecasts impending liquidity squeezes, the current disposition behavior is automatically cut back by negative feedback. This means that not all current

disposition processes are realized and the impending liquidity squeeze can be avoided. In compliance with the banks' rating guidelines the software agent ensures that

- no funds are allocated before their fixed value date
- all cash clearing or cash pooling activities are always performed in such a way that the corresponding accounts are not overbooked with regard to their near future (e.g. direct debiting). It is precisely the lack of consideration of future direct debiting procedures that causes severe problems in conventional cash management systems, so that the interest gains through cash pooling are lost again by overdrawing other accounts with higher current account interest.
- the credit lines of the current accounts are not overdrawn and that automatic transactions are triggered on time for direct debiting, leading to a liquidity spot landing for the respective account.

Example: Based on the bank entry evaluations the software agent realizes that a credit entry amounting to 100,000 € has been booked but the value date was fixed only in 3 days time. Correspondingly, disposition of due liabilities will automatically be shifted by 3 days so as not to violate the banks' rating criteria for account management by the disposition of funds that have not reached their value date.

7. Consistent realization of optimization potential

Investment and borrowing strategies are activated by the software agent on a daily evaluation basis and no longer on a monthly evaluation basis as before..

This procedure allows for the achievement of short-term interest gains without threatening future liquidity development.

Example: The agent system proposes an investment sum of 50,000 € in 2 tranches, in accordance with the liquidity forecasts calculated, one for 10 days and one for 30 days at 60% and 40% respectively. If the money is invested as a fixed deposit the interest gain achievable is about 140 €, which is earned as an additional revenue within a daily financial optimization. Even if the interest gain seems marginal at first glance, it will accumulate throughout the year to about 20,000 € due to the strategy of applying numerous small optimization steps.

Feature	Planning to date	Software agent
Planning period	month	day
Perception	retrospective	prospective
Planning methods	Details, systematic, reducing and standardizing, systemic errors	Interactions, systemic, extending associatively, correct basic pattern
Effort	+	-
Precision	+	-
Accuracy	-	+
Proactive disposition	-	+
Optimization	-	+
Automatic regulation	-	+
Learning	-	+
Prospection	by humans	by computers

1 Rating S

1.1. Company ranking

Type of company: Commercial customer

Sector: Other

1.2 Financial rating evaluation

Ratio		Weighting		Rating
				Grade (1- 15)
Financial status				8
Equity ratio	32%	24%		
Liquidity ratio	0%	10%		
Capital commitment	8%	8%		Overwritten
Outside capital structure	81%	13%		---
Profit situation				
Gross profit situation	8%	11%		
Interest expense ratio	0%	0%		
Rent expense ratio	0%	0%		
Financial situation				
Cash flow ratio 1	28%	13%		
Dynamic operating profit	16%	13%		
Creditors' terms	19	8%		

1.3 Warning signs

The business account was overdrawn more than twice in the past 12 calendar months, for longer than two months on each occasion.

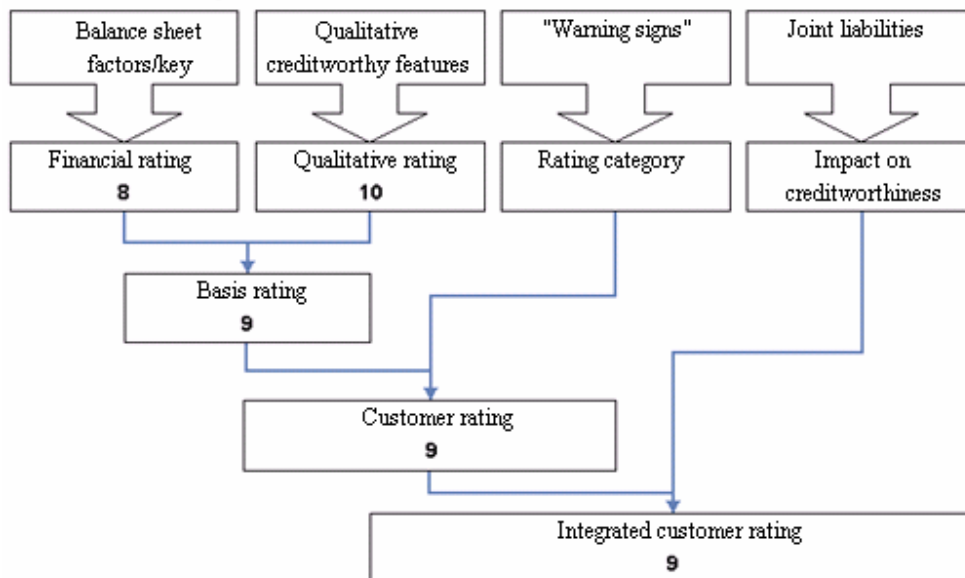
There was a one-off default in the past 12 calendar months, lasting two months or longer.

There were no check or debit note returns in the past 12 calendar months.

1.4 Joint Liabilities

There are no joint liabilities.

1.5 Overall rating S



1 Rating S

1.1. Company ranking

Type of company: Commercial customer

Sector: Other

1.2 Financial rating evaluation

Ratio		Weighting		Rating
Financial status				Grade (1- 15)
Equity ratio	32%	24%		8
Liquidity ratio	0%	10%		
Capital commitment	8%	8%		Overwritten
Outside capital structure	81%	13%		---
Profit situation				
Gross profit situation	8%	11%		
Interest expense ratio	0%	0%		
Rent expense ratio	0%	0%		
Financial situation				
Cash flow ratio 1	28%	13%		
Dynamic operating profit	16%	13%		
Creditors' term	19	8%		

1.3 Warning signs

The business account was not overdrawn in the past 12 calendar months.

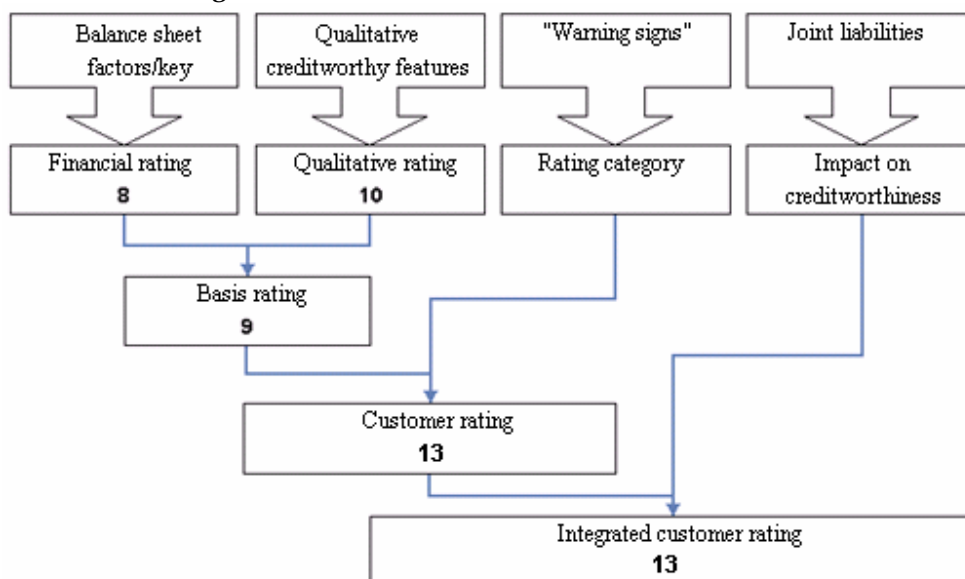
There were no defaults in the past 12 calendar months.

There were no check or debit note returns in the past 12 calendar months.

1.4 Joint Liabilities

There are no joint liabilities.

1.5 Overall rating S



Results for the company

Using the software agent will lower interest costs in the 13% to 9% rating. The substantial gain in transparency in the company's financial engineering will guarantee entrepreneurial safety. The return on sales this financial year is expected to be 3.4%, i.e. it will exceed that of last year by 0.4%. There will be a 20% drop in the use of the credit line. The interest costs saved as a result will amount to 50,000 €.

In addition, interest revenues through short-term investment strategies will amount to about 20,000 €. Correspondingly, a reduction of the current account overdraft limit is planned for the coming financial year.

Author:

Dr. Mathias Künlen, CEO Softmark AG, Hirtenweg 2, 82031 Grünwald, Germany
(Internet: www.softmark.de, email: mk@softmark.de)