Retail Simulation Engine | Price Engine

product & concepts


Status March 2008 | Release 1.6

Table of Contents

There is room for improvement in the pricing of many articles in retailer assortments. The Price Engine from Dacos® allows retailers to improve their gross profit through tactical price optimization in the permanent price sector. The goal of tactical price optimization is to increase the average gross profit per transaction, without changing or damaging a company's price image. This white paper describes the principles, technical requirements and user potential for this effective form of price optimization.

## Introduction

Spectactular sales and discounts ("20% off everything except pet food") belong to the more aggressive and today, widespread marketing instruments in the German retail industry. Unfortunately, the oversupply of short-term teasers trains many customers to become bargain hunters with low price acceptance in the permanent price sector. A retailer who focuses on such methods only risks damaging the image and success of his business on a long-term basis.

While discounters and wholesalers continue to achieve satisfactory returns, small and middle-sized supermarkets make too little. Many retailers neglect the permanent price sector for short-term promotional sales to distinguish themselves from the tremendous discounter competition. However, most German commercial enterprises achieve a profit margin of less than 1% in this manner.

Profit margin often less than 1 percent

Discounts and temporary teasers are meaningful for increasing customer frequency and securing sales but, the vast majority of the earnings in the German retail industry continue to be achieved with standard prices. Thus, potential for higher profits lies in the meaningful coordination of customer rebates, temporary sales and a good, long-term price image in the permanent price sector.

Potential for higher profits in the permanent price sector

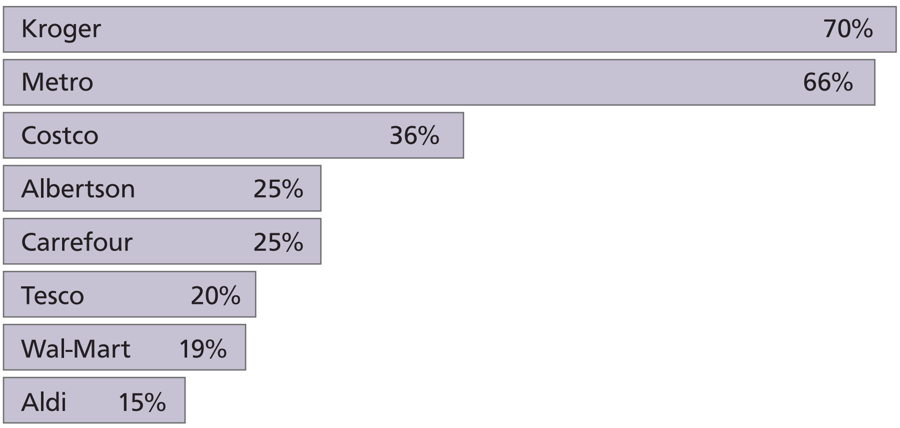



Figure 1: A 1% price increase leads to the following increases in yield [1]

Sales price increase is the best way to increase profits

The influence of prices on a retailer's profit is more direct than that of cost reductions or increases in sales volumes. Figure 1 shows the effect of a 1% price increase on the entire assortment of several large chain stores (assuming that the sale down remains constant). For many retailers an investment in improved price finding processes is the best way to increase profits [1].

### Using Assortment Competence in the Price Game

Using assortment competence in the price game

In well-sorted categories one finds "key articles" that are especially important for the positive price perception of the categories, paired with less prominent articles targeted at customers with varying quality standards and habits. Playing a clever, tactical game with the prices of the A- and B-brands, the often especially price-sensitive store brand and the peripheral assortment is very difficult however, it presents chances for increasing returns.

Articles offered in article groups are evaluated by customers according to quality characteristics and price. Price changes can influence purchasing decisions, particularly in cases where the price is the high-priority decision-making criterion between the various alternatives. But how does a quality characteristic such as "organic" effect purchasing decisions in a large group of articles with many alternatives? What impact does the price increase for a brand-name organic product in the higher price sector have in contrast to a price reduction for your own organic store brand? Does it make more sense to reduce the price of a key article with a high market share or to reduce several prices in the peripheral assortment?

The answers to such questions are highly valuable for achieving a long-term increase in returns through tactical assortment pricing measures without creating negative price perception on the customer's side. Unfortunately, these answers can't be found efficiently without the appropriate tools. Full-line distributors and power stores offer between ten and one hundred thousand items. Product categories with several dozen alternative product groups aren't rare here. No retailer has the resources to define optimal pricing for every category by carrying out complicated surveys and conjoint analyses, especially because pricing is not a one time, self-contained process: competition never sleeps and manufacturers are constantly adjusting their prices as they see fit. The solution to these problems is a tool with a high degree of automation that can forecast a customer's reaction to price changes and determine appropriate prices.

IT-tool is the only possible solution

## What is Tactical Price Optimization?

The **Tactical Price Optimization**, the core of the Price Engine, simulates and optimises price changes in the assortment's permanent price sector. On the basis of condensed sales data from several months and current purchase prices, the Tactical Price Optimization determines a mathematical model of the psychological dependencies between retail prices and buying probability on the one side and the expected gross profit per sales unit on the other. With this model, changes in buying behaviour following price modifications can be simulated and ideal recommendations for price lines automatically created. The Tactical Price Optimization is a practice-oriented tool that observes the laws of the market and retail economics:

Simulate changes in buying behaviour after price modifications

- ▲ The price simulation calculates exact estimates of the gross profit per sales unit and basic amount. The user can thus, quantitatively compare alternative modification recommendations and by doing so, determine the best method for a category.
- ▲ The Price Optimization maintains and improves the positive overall perception of the prices in a product line. "Extreme" price changes, especially those in key articles, are not recommended. Price increases for some products are automatically balanced out by beneficial price cuts in alternative products.

Price optimization maintains overall perception of the prices

- ▲ Price limits for products can also be defined manually with minimum and maximum specifications, so that changes can be prevented where they are not intended.
- ▲ Important psychological price thresholds and a company's price policy ("all prices over a Euro end with a 5 or a 9") are not just taken into consideration, but used to their full capacity.

Maintain price policy

- ▲ The influence of different packaging sizes as a quality characteristic and a determining factor for price sensitivity and possible stockage also enters into the model.
- ▲ Unusual sales processes ("giant customers", extreme one-day discounts etc.) are statistically identified and automatically filtered out of the data.
- ▲ Prices can be optimised based on regional differences. To do this, the price model from the Tactical Price Optimization uses different customer preferences and given differences in product selection to calculate regionally optimum price lines.

Zone pricing possible

The gross profit for almost all product lines can be increased by an average of two to four percent by simply optimising permanent prices. Untapped potential for even more gain can especially be found in neglected product lines belonging to the peripheral assortment. The Tactical Price Optimization helps you increase your return from many different product lines with a well thought-out price policy.

Increase in gross profit by 2 to 4 percent

DACOS® SOFTWARE

### Further Components of the Price Engine

The Price Engine analyses all hierarchical levels in your range of items down to bundled and single items. In addition, it also determines important key figures for brands and suppliers. Complete reporting views are available. These cover for example, the retail price, sales volume, sales and return-histories, as well as listings, delisting and promotional campaigns.

Integrate analyses into the engine

Assortment managers and marketing departments can embed all of the tactical price measures into the strategic tasks of category management based on these analyses. The available assortment analyses allow you to consider tactical pricing measures in contrast to alternative measures in the assortment policy, such as for example, listings or delistings.

## Customer Models for Optimizing Prices

Classic optimization per price-demand function of individual articles

Classic approaches to the optimization of prices usually start with assumptions on the connection between the price and the sale down quantity of individual articles. In the simplest case, the sale down behaviour of articles is understood as a function of current prices. If a price-demand function exists then you can determine a price that allows the retailer to achieve the maximum return with relatively simple mathematical means [2].

Unfortunately, in practice, such a price function can only be determined for a few articles and certain types of markets. In other cases, one may stumble across the following difficulties:

- The price change data available from the past is usually inadequate for making statistically secure estimates on the price elasticity of individual articles. If for example, an article was always sold at a certain price, it is impossible to make secure assumptions for this individual article with regard to the reaction of the sale down on a price cut or increase.
- An article's price is just one of many influencing factors. Seasonal fluctuations, the weather, advertising campaigns, cannibalisation and push by substitutable articles etc. often influence sales much more strongly than customary price changes in the permanent price sector.
- Traditional assumptions on price functions often prove inadequate under real-life conditions. For example, price elasticity usually changes in a manner other than predicted depending on the article group when prices are increased. Thus, even small errors in the assumptions regarding the price function can provide false conclusions about the sale down. The result is "optimum" prices that are incorrect.

### The Dacos Approach: Simulation with Price Models for Article Groups

New approach works from the customer´s point of view

To avoid the problems described above Dacos uses a revolutionary approach, which considers the price optimization problem from a different point of view, namely, the customer's view. In many groups of articles the customer has the choice between different, but largely interchangable products. A customer's buying behaviour, i.e. the respective share of alternatives bought in a group of articles over a certain period of time, provides a clearer picture of the connection between prices and buying decisions, as the consideration of isolated articles does. Thus, models of the buying behaviour typical for customer groups in article groups with substitutable products form the foundation of the Price Engine's tactical price optimization. The question is: how do customers react to price changes and how can retailers profit from this?

Model of price elasticity

First, the Price Engine creates a complex mathematical model based on the current condensed sales data from several months that estimates the relative price elasticity of all substitutable alternatives and thus, the customer's willingness to change their buying behaviour as a result of price changes.

This customer model indirectly predicts how the sale down of a product changes in comparison to other substitutable articles when the retailer changes prices long-term. The ability to predict price elasticity is also of practical relevance when price changes occur infrequently or not at all within a certain period.  If price changes occur within the model's learning period you can use them to improve the accuracy of the prediction.

A relative, networked model of price elasticity can be used directly for a profit and loss statement, which results in a **value for the average return per sales unit**. Of course, this value depends on a range of purchasing conditions. Thus, such a profit and loss account re-quires suitable purchase resp. cost prices, in addition to retail prices and estimated market shares.

Unlike conventional price models whose estimates are targeted at clearly measurable absolute quantity changes, this relative model is accurate enough to estimate the return-relevant effects of small price changes. The optimization of the return can therefore be made on the basis of subtle price manipulations "below the radar" of customers and the competition. Aggressive price changes whose effects can only be predicted with difficulty are avoided. A further advantage of such a relative price model across a complete group of articles is that the predictions made are not dependent on seasonal fluctuations.

Price change "below the radar" of customers

**The Tactical Price Optimization's goal is always an increase in the return of a group of articles without fundamentally interfering with the salesw of all products: relatively small price changes are used to achieve a few percent more return for each group of articles.**

Increase in return without interfering with the sales of products

## An Example: The Optimization of a Small Group of Articles:

The following example shows how the tactical price optimization functions based on the example of a small group of articles consisting of four different baby shampoos. The following table shows the retail prices, gross profits and market shares (purchasing probability) for each of the shampoos:

| Article | Price in Euro | Gross profit in Euro | Market share in % |
|---|---|---|---|
| Article A | 1,42 | 0,02 | 35,7 |
| Article B | 1,71 | 0,54 | 23,2 |
| Article C | 1,83 | 0,45 | 17,1 |
| Article D | 3,78 | 0,92 | 24,0 |

Article A is an eye-catcher article. Its average price in all of the stores is 1,42 Euro, the average gross profit is only 2 Eurocent per package. The market share of article A is 35,7%.

Article B costs an average of 1,71 Euro and has a market share of 23,2%. Article C costs an average of 1,83 Euro and has a market share of 17,1%. Slightly more is earned with article B than with article C, whose margin is 0,45 Euro despite article B's slightly lower retail price of 0,54 Euro.

Push articles with high gross profit

Article D is comparatively expensive at 3,78 Euro. The gross profit is 0,92 Euro, the market share 24%.

The gross profit for all four articles comes to an average of 0,396 Euro. The package sizes are the same (150 ml), the effect from the difference in the range of consumption of different package sizes can therefore be ignored.

People that buy products in these article groups evaluate all of the alternatives with the help of an abstract benefit. The price being paid belongs to this abstract benefit. If prices change then the individual consumer acceptance for an alternative can change for each customer: a price cut in one of the four products may suddenly make this product the most attractive alternative for some of the customers that had previously bought another product. If you can predict the effect that the price will have on the all of the customer's willingness to pay then the average gross profit can be changed in favour of the retailer without significantly varying the sales volume. Concretely, three of the four shampoo prices were slightly changed for the model-supported tactical price optimization of the four shampoos.

Fully utilise customer´s willingness to pay

| Article | Price in Euro | Gross profit in Euro | Market share in % |
|---|---|---|---|
| Article A | 1,45 | 0,05 | 35,0 |
| Article B | 1,65 | 0,48 | 24,7 |
| Article C | 1,83 | 0,45 | 23,8 |
| Article D | 3,85 | 0,99 | 16,6 |

The price for article C remains unchanged. The price for article B was reduced slightly. The price for the high-quality, brand-named article D with low price elasticity was however, increased.

Due to the changes in price some customers switched from article A to article B, because the prices for both were now more similar and B appeared more attractive because of its somewhat lower price. In addition, some customers switched from article C to article B because the price gap between these articles was now larger. The gross profit for article B is higher than that of article C, which influences the entire gross profit positively.

The market share for article D decreases; its gross profit however, increases.

The gross profit for all four articles increases by 3,3% to 0,4059 Euro. All in all, only a few customers decided to buy an article other than the one they bought before the price change. A measureable improvement in returns however, was achieved due to these small customer movements.

## The Automatic Improvement of Prices

Simulation based on a model for buying-behaviour

A manual simulation and evaluation of price changes is very complicated in article groups with many alternatives even when an appropriate computer-supported model for buying behaviour is available. It is, for example, only possible in a very small case study, such as the above, to interactively analyse all of the conceivable price alternatives with the help of a model. In reality however, a category manager never has the time to play out all of the price changes possible for all of the article groups he is responsible for. In a single article group with only 10 alternatives and 5 price alternatives per product (for example: two price cuts each on the next lower price threshold, two price increases on the next higher prices and the alternative of keeping the current price) one would have to try out 9,765,625 different price alternatives and then evaluate them with regard to their effect on the company's price image.

Thus, the Price Engine can also **automatically optimize prices**, in addition to allowing the manual simulation of price changes. The optimization generates modification recommendations based on current retail prices that can achieve the following goals:

- All of the recommended retail prices are on the retailer's price scheme resp. that of the article group (for example: all of the prices end with a 5 or a 9).
- The average gross profit per consumed unit sold should increase.
- On the average, the customers affected by the price changes should see balanced price cuts and increases; the yield increase shouldn't result from price increases alone.

Optimization with balanced price cuts and increases

- The gross profit should increase slightly, even when fewer customers make other buying decisions due to price changes than predicted by the model. The optimization thus, minimises the risk represented by price cuts.
- The number of price increases should in total remain low; the ratio of price increases to price cuts should be positive for the company's price image.
- The amount of price changes for individual articles should remain as low as possible; in particular for example, price increases on strong-selling key articles are avoided as far as possible.

Only small price changes

The assortment manager can efficiently improve the system's recommendation with manual intervention possibilities (see section User Interface). Thus for example, price limits (minimum price, maximum price) can be defined if the recommendations for certain products diverge too strongly from the assortment's current price policy. With version 1.6, the Retail Simulation Engine supports the integration of competitor prices collected by the retailer for defining price strategies.

## Price Optimization with Genetic Algorithms

The task of automatically creating price recommendation described in the previous section is an optimization problem that is difficult to solve. The high number of possible price combinations and the combination of several optimization goals – yield increase, a well-balanced price image and risk minimization – excludes the use of many conventional optimization methods from the field of operations research. Linear optimization methods, such as the

Simplex-algorithm for example, fail due to the fact that the connection between prices and market shares for several products is always non-linear in a realistic model of the buying behaviour.

The Dacos Simulation Engine uses techniques from the field of Artificial Intelligence, in particular machine-learning methods on the basis of genetic algorithms, to create customer models and optimize price recommendations.

Optimization uses methods of Artificial Intelligence

A **genetic algorithm** is a heuristic optimization method used to solve complex problems where no appropriate quick analytical solution exists. For genetic algorithms, first a large set of random solutions is created, the so-called "initial population". The best individuals in this population are selected and combined according to a quality factor, similar to the evolutionary process. The resulting "children" then represent the next generation of the population and will often have a better average value with regard to the target function than the previous generation because usually "fit" parents create even better children. Individuals whose quality factor is below average are given fewer chances to "reproduce" and a limited habitat makes sure that they eliminated over time.

Adaption to real customer´s behaviour with genetic algorithms

This procedure of selection and combination – **the evolutionary search** – is carried out until no more improvements are observed. Incidental mutations – small changes in an individual – serve to minimize the probability of stagnating at a local maximum.

## Evolutionary Search

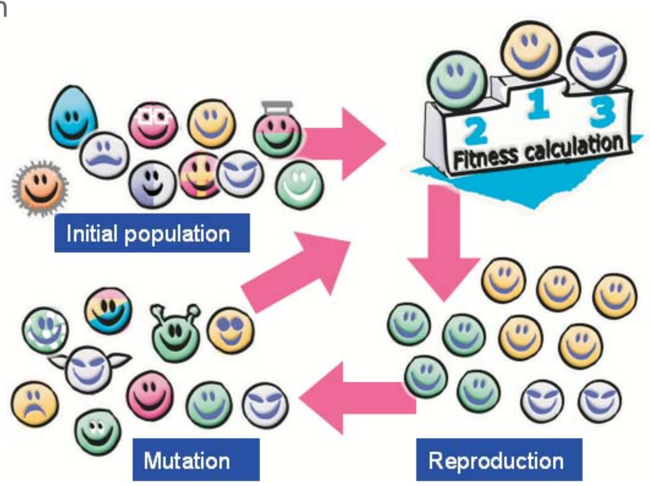



Figure 2 describes the course of a genetic algorithm.

In the case of the automatic price optimization, the population of the genetic algorithm consists of random price changes. With the customer model you can calculate the simulated return per unit for each "living creature" and the degree to which other auxiliary conditions are fulfilled, such as for example, the use of largely moderate price increases compared with great changes. Thus, each individual receives a fitness that decides whether it will survive and have the chance to reproduce.  The Dacos Simulation Engine can calculate hundreds of generations with thousands of price recommendations within just a few minutes on a standard server. The end result consists of price recommendations that generate a higher gross profit with respect to the price images as the initial prices do.

## User Interface

Intuitive user interface

The intuitive user interface from the Dacos Simulation Engine allows the user to efficiently carry out price optimization projects. The Price Engine's tactical price optimization process comprises the following steps:

1. The creation of a new price project or the update of an earlier project.
2. The definition of essential parameters.
3. The selection of suitable products.
4. Data retrieval and the exclusion of unsuitable products.
5. The adjustment of price limits.
6. The creation of optimization results.
7. Further adjustments to price limts and a re-optimization or
8. The export of results to a merchandise management system or other suitable interfaces in the operative system.

The following steps provide you with an overview of the possibilities a user is given to achieve the desired results with the Price Engine.

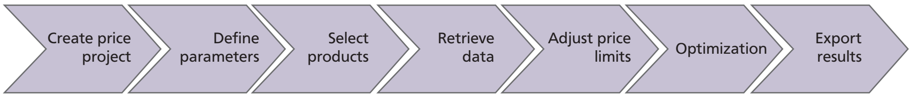



Figure 3: the integrated price optimization process with the Dacos Price Engine

### Creating a New Price Project

The Retail Simulation Engine carries out all simulations and optimizations in the form of so-called use cases. These store all of the parameters and intermediate data, so that they are suited for the management of continual processes such as for example, the regular maintenance of prices and assortments. The stored results can also be used for the subsequent control of the predicitions made.

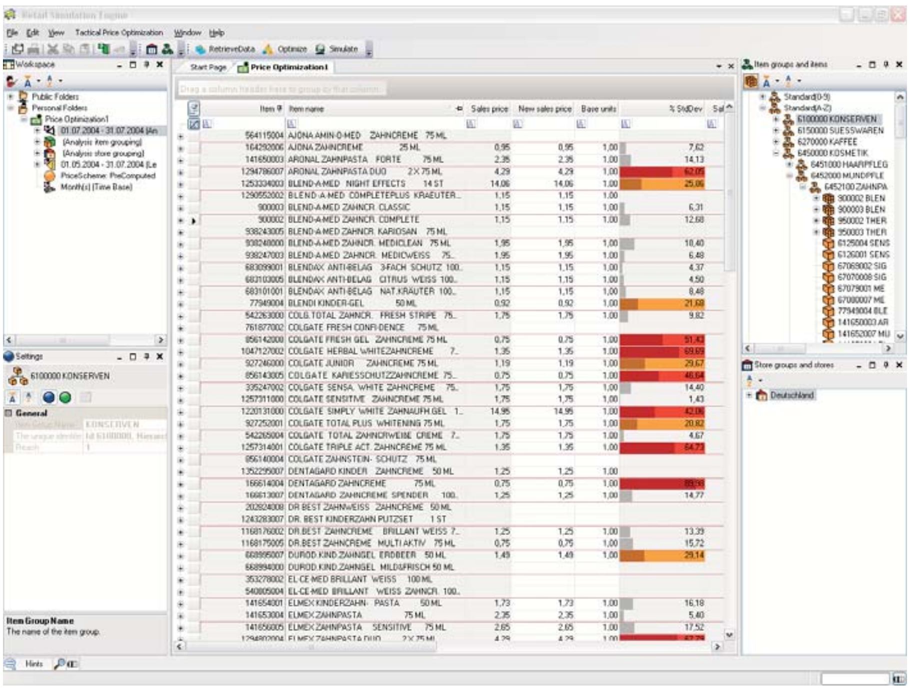


Figure 4: an overview of the user interface

## The Definition of Essential Parameters

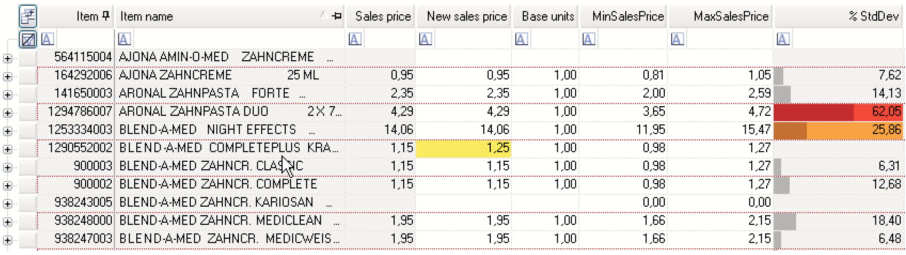
The Tactical Price Optimization uses the condensed sales data from the article groups to be analysed based on a defineable group of branches, for example, centrally for all branches or regionally for regional prices. The essential parameters of a price optimization are: the start and end-date of the **learning period**, the **reference or analysis period** for the evaluation of the results and the selection of **the branches being observed**. You can also select a schema with the approved target prices ("all prices end with a 5 or a 9, prices such as 1.05 or 1.09 are however excluded") for the automatic price optimization. Logical default settings are defined for all parameters so that an assortment manager can quickly start product selection.

Model uses sales data from the past

## The Selection of Suitable Products

The user can quickly navigate to the products relevant for him over the standard hierarchy and then add these to his work area per drag and drop. An article selection can be made up of several article groups in so far as they consist of substitutable articles (for example: washing detergent in powder or tablet form). A combination made once can be used as a part of a use case, but can also be deposited as an individual object in the work area and is thus, available to the assortment manager at all times for further work. The product hierarchy considers logical article groups that must carry the same price **(a collection of articles)**. Article collections are always handled as individual articles by the price optimization interface.

Choose substitutable articles



| Item | Item name | Sales price | New sales price | Base units | MinSalesPrice | MaxSalesPrice | % StdDev |
|---|---|---|---|---|---|---|---|
| 564115004 | AJONA AMIN-O-MED ZAHNCREME ... | | | | | | |
| 164292006 | AJONA ZAHNCREME 25 ML | 0,95 | 0,95 | 1,00 | 0,81 | 1,05 | 7,62 |
| 141650003 | ARONAL ZAHNPASTA FORTE ... | 2,35 | 2,35 | 1,00 | 2,00 | 2,59 | 14,13 |
| 1294786007 | ARONAL ZAHNPASTA DUO 2X 7... | 4,29 | 4,29 | 1,00 | 3,65 | 4,72 | 62,05 |
| 1253334003 | BLEND-A-MED NIGHT EFFECTS ... | 14,06 | 14,06 | 1,00 | 11,95 | 15,47 | 25,86 |
| 1290552002 | BLEND-A-MED COMPLETEPLUS KRA... | 1,15 | 1,25 | 1,00 | 0,98 | 1,27 | |
| 900003 | BLEND-A-MED ZAHNCR. CLASSIC | 1,15 | 1,15 | 1,00 | 0,98 | 1,27 | 6,31 |
| 900002 | BLEND-A-MED ZAHNCR. COMPLETE | 1,15 | 1,15 | 1,00 | 0,98 | 1,27 | 12,68 |
| 938243005 | BLEND-A-MED ZAHNCR. KARIOSAN ... | | | | 0,00 | 0,00 | |
| 938248000 | BLEND-A-MED ZAHNCR. MEDICLEAN ... | 1,95 | 1,95 | 1,00 | 1,66 | 2,15 | 18,40 |
| 938247003 | BLEND-A-MED ZAHNCR. MEDICWEIS... | 1,95 | 1,95 | 1,00 | 1,66 | 2,15 | 6,48 |

Figure 5: the input window

## Calling up Data and Excluding Unsuitable Products

All of the relevant key figures are called up from the data warehouse for the selected articles and article collections according to the parameter settings. The package sizes are also called up in order to consider the different consumption quantities in the price optimization, in addition to prices and sale down quantities. The Price Engine automatically corrects irregularities in the sale down, due for example to sales or "giant customers" and in addition, displays the relevant key figures that allow the user to identify unsuitable products. Thus, for example, you can reliably identify and eliminate products already delisted due to their falling sales numbers. The following article data is displayed:

Exclude irregularities

- Article – the article number.
- Article description – the name of the article.
- New sales price – the retail price selected for a price simulation, preset to the known retail price from the analysis period.
- New purchase price – the purchase price selected for the price simulation, preset to the known purchase price from the analysis period.
- Gross profit – the gross profit per article, i.e. the difference between the (net) retail and purchase resp. the cost price.
- Base units – the number of items sold within the learning period, i.e. transactions. An optional entry, Adjusted Quantity, shows the value after filtering out advertising effects.
- MinSalesPrice – the lowest retail price permissible for the optimization. The optimization may select a somewhat higher price than the lowest retail price, when the MinSalesPrice does not correspond with the preset price schema.

- MaxSalesPrice – the maximum retail price permissible for the optimization. The optimization may select a somewhat lower price than the maximum retail price, if the MaxSalesPrice does not correspond with the preset price schema.
- Pack size – the package size of an article in the base quantity units. The basis for the quantity unit is defined by the retailer and can be entered in pieces and units of weight (for example: á 100 g) or for example, in the number of laundry loads per universal washing detergent.
- % StdDev – the mean error is the average deviation of the sales figures per month from the average of the learning period. It helps to identify articles with sales figures that fluctuate heavily, as often occurs with listings and delistings. Values above 20% together with high sales volumes are a clear sign for sales or changes in the listing.
- # Stores – the maximum number of stores where an article was sold within the learning period.

The sales data per month can also be viewed in detail.

The Tactical Price Optimization can only be used on alternatives available to the customer in roughly the same manner and actually considered as alternatives by the customer. Articles should be removed from the selection when one of the following conditions is fulfilled:

- The article was sold in fewer branches as the other alternatives.
- The mean error is very high, because for example the article was recently listed or is being delisted. Values over 20% for articles sold often can be an indicator for listings or delistings.
- The package size differs from the other alternatives so much that a substitutability cannot be assumed – thus for example, customers usually do not select samples and travel sizes as alternatives to normal package sizes.

All of the alternatives that do not fulfill these conditions should, vice versa, be kept in the selection.

## The Adjustment of Price Limits

Observe price limits

The possible retail prices for alternatives are often subject to restrictions that defined by the retailer's price policy and contracts with suppliers and manufacturers. It is wise to adjust the price floor and ceiling, provided that changes in the retail and purchase prices are automatically optimized. This adjustment can however, be made subsequently after a first fully automatic recommendation is made. Competitor prices, if known and stored in a data warehouse, can also be used as a basis for price limits in version 1.6 of the Retail Simulation Engine. For example: the maximum prices are then limited by a competitor's current prices.

## The Creation of the Optimization Results

Optimization at the push of a button

A click on the button "Optimize" starts the evolutionary search for beneficial price changes. The calculation of a buyer model and the optimization may take several minutes depending on the processing power of the server being used and the complexity of the article group however, this period is shorter for interactive changes, for example subsequent changes to price limits. The following figure shows a summary of an optimization process.

| Result type | Contribution/U | Contribution | Cost of sales/U | Ø Sales Price/U | Ø Sales Price Change/U | % Level/U | % RMSE |
|---|---|---|---|---|---|---|---|
| Anaylsis | 0,6429 | 0,6872 | 0,7333 | 1,6377 | 0,0000 | -0,1 | 0,000 |
| Model | 0,6436 | 0,6878 | 0,7294 | 1,6339 | 0,0000 | 0,0 | 0,003 |
| Simulation | 0,6616 | 0,7040 | 0,7381 | 1,6656 | -0,0001 | 2,8 | 0,000 |

Figure 6: the input window (summary)

There are three different types of results for a tactical price optimization:

- Analysis – the information from an analysis period selected for reference purposes. This information allows a comparison with the simulation of the normal prices (the model) and thus, an estimation of how well a model really meets the reality of the analysis period.

- **Model** – the mathematical model of the buying behaviour is applied to the prices in the analysis period.
- **Simulation** – shows the information resulting from the simulation of optimum retail prices or prices set by the user.

The table summary displays the following information:

- **Ø Contribution/U** – the weighted mean of the gross profit per base unit sold. Base units result from the package size in which the observed article is sold.
- **Δ Contribution/U %** – the difference in percent between the weighted gross profit per unit and the weighted gross profit of the model. This value shows the expected increase in the gross profit per unit in the line Simulation however, under the item Analysis it shows the error the learned model exhibits for the analysis month.
- **Ø Contribution** – the weighted gross profit per article sold without normalization on the base units.
- **Ø Purchase price/U** – the weighted purchase price per unit sold. The value is irrelevant for normal use of the Tactical Price Optimization and can therefore be ignored.
- **Ø Sales price/U** – the retail price paid per unit by the customer in the weighted mean.
- **Δ Sales price/U** – the price difference that can be observed for the total number of articles sold in the weighted mean of the sale down. Negative values mean that the prices were probably reduced for all alternatives. Customers can pay a higher price at Ø Sales price/U despite such a price cut if for example, they are animated to buy more expensive products through clever price cuts.
- **RMSE** – the root of the mean squared error of the estimated market shares gegenüber the analysis period. This value is, similar to the Δ Gross profit/U % in the line Analysis an indicator for possible problems in the data the model is learned from, for example, fluctuations in the sales numbers due to sales or listings.

Automatic assessment of the optimization quality

The central value for the assessment of the optimization is Δ Gross profit/U %. In the line Analysis, this value states how well the learned model “fits” the analysis period, i.e. how accurately the model can estimate the actual weighted gross profits. Values below 3% point to a usable model that was learned from a learning period without dramatic fluctuations in the sales of important alternatives. Excellent models carry a value below 1%. If the value under Analysis is acceptably low, then a high value from Δ Gross profit/U % means that the calculated price change has a higher chance of triggering a considerable increase in the mean gross profit of the alternatives. The closer the estimated increase in the gross profit is to the determined model error, the lower the expectable actual increase in the gross profit is. In extreme cases, the increase in the gross profit may be lower than the model error. If this is the case, then the article group is not suited for an automatic tactical price optimization on the basis of the data from the selected learning period.

The model’s error value is based directly on the difference between the model’s estimate of the market shares and the data from the analysis period. Values below 0.01 are generally acceptable however, higher values invalidate statements made on the effect of the price changes.

The detailed information on the analysis, model and simulation results contains the following:

- **Δ Sales price** – the change in the article’s retail price compared with the analysis period
- **% Quantity** – the quantity share (market share) an article has in the total amount sold. This value is measured for the analysis period. However, for the model and the simulation it is calculated according to the selected or optimized prices from the learned model. It is not normalized on the sales unit (base units).
- **% Profit** – the share of return in percent an article has in the total return of the alternatives. Similar to the quantity share, this value is measured for the analysis period. For the model and the simulation however, it results from the prices set and the market shares that the model calculates from this.
- **% Quantity/U** – this value states the quantity share based on the base units, analogue to the unnormalised market share.

- ∆ Gross profit/U – the difference in the gross profit per unit against the analysis period. This value is always zero in the analysis and the model.
- ∆ Quantity/U % – the change in the quantity share (market share) compared to the model in percent. This states the shift in the quantity share that will be triggered by the price changes for the simulation. Under the item Analysis however, this value states the model's estimated error for the article.

As long as the article, package sizes and the project's general parameters (learning period, analysis period and branch grouping) aren't changed, the user can modify price limitations as desired and by doing so, quickly generate new results because the calculation of a new model isn't necessary.

### Manual Simulation

The automatic optimization of prices results in a price recommendation that may contain a series of price changes. The manual simulation of price changes is also avaliable as an alternative if you wish to try out the effect of a limited number of known price changes or you only want to realise a part of the change recommendations.

User can interfere anytime

You can select any retail and purchase price after selecting the corresponding alternatives and carrying out the data request. Similar to optimizings, you can also change the basic units of the article, in order for example, to adjust incompatible package sizes. The values from MaxSalesPrice and MinSalesPrice are however, ignored, in contrast to an automatic optimization.

By clicking the button Simulate you can generate a simulation result that corresponds with an optimization in the form of its result. The information in the line Simulation now however, does not display the simulation of the optimum retail prices, but rather a simulation of the prices defined manually by the user. As long as the article, the package size and the project's general parameters (learning period, analysis period and branch grouping) are not changed, the user can change prices in any manner he sees fit and generate new results quickly because a new model doesn't have to be calculated.

Simulation at the push of a button

## The Advantages of the Price Engine

The Price Engine is based on the most up-to-date findings in Artificial Intelligence and retail and marketing research. It combines powerful machine learning methods with robust econometric behavioural models to realise a practice-oriented procedure for increasing a retailer's profit.

Practice-oriented procedure

The Price Engine's Tactical Price Optimization comprises two functions:

- The creation of price change recommendation for selected alternatives; i.e. the optimization of prices
- The simulation of the effects resulting from changes the retail and purchase prices of individual articles, with regard to the entire group of articles.

The simulation and the optimization is used to define small price changes ("below the radar of customers and competitors") that change little in the total number of units sold, but can generate significant profit-enhancing effects through switchers.

In summary, the use of the Price Engine offers the following benefits:

- The gross profit is improved by at least 2% in many product ranges.
- The evaluation of price optimizations from approx. 20 randomly selected groups of articles from the sectors Food, Near Food and Toiletries has shown that increases in the gross profit of an average of 4% can be realised without affecting the price image. This would mean an increase in the gross profit ratio (gross profit / turnover) of 0,6 percentage points for an average German retail company with a gross profit ratio of approx. 15% (Source: Federal Statistical Office 2003).
- With steady costs this would lead to a direct increase of 0,6% in the annual operating margin before taxation. With the Tactical Price Optimization an average retail establishment can achieve a profit increase of up to 6 million euros per annum based on a price-optimized assortment section with annual sales of 1 billion euros.

Increase in gross profit by 2 to 4 percent

- The margins are significantly improved in article groups with many substitutable articles. Profit can be made here by retailers with assortment competence who want to offer their customers a wide selection with different prices and levels of quality.
- The assortment manager has many manual possibilities for intervention in order to harmonize the company's price policy with the automatically generated recommendations. The software supports the assortment manager in achieving a consistent price policy for the assortment that is recognizable by the customer.
- Price changes are tested by simulating their effect on the article group. The model's automatic comparison of the customer's reaction to price changes with past data allows the system to make high-quality statements.
- If desired, the price structure within article groups can be balanced, i.e. the number of varying prices within an article group is reduced.
- Price lines are maintained by communicating prices from the head office to the branches. Different prices for the same product, depending on a region's buying power, remain possible.
- The price image is maintained because recommended price changes are minimal and remain below the customer's and the competition's radar. This means you can retain your customers without animating price wars with the competition.

Price image is maintained

- Price changes in the industry can be simulated ahead of time and statements made on price elasticity and cannibalization. Thus, the Price Engine also supports category management in meetings with suppliers.
- Active category management is promoted. The article groups are developed so that the retailers, as well as supplier's goals are considered.
- The simple operation of the Price Engine with its clear-cut menu structure and screen design provides a high degree of usability and thus, allows the efficient application of experts in category management.
- The implementation is simple and can be carried out easily. The length of a typical project from the first joint workshop to the last user training session is under three months.

Implementation is easy

- The Retail Simulation Engine is SAP Netweaver-certified.

## Requirements for Use

As an application of the Retail Simulation Engine, the Price Engine is part of a modern, scalable software architecture. In addition to the installation of the Retail Simulation Engine on a suitable application server and the GUI on the user's workstation, the Tactical Price Optimization requires access to a data warehouse containing the relevant data. Today, usually all of the information required is already available there:

- Sales data condensed daily over a period of several months including all of the average retail and current purchase prices.

Solely daily condensed sales data are required

- The article group hierarchy for the navigation including information product groups with the same pricing
- Information on package sizes and base consumed units
- Information on promotional measures

Further important information for the configuration of the Tactical Price Optimization such as for example desired price thresholds and the retailer's price schema will be collected together with our Dacos-consultants in the course of a consultancy project.

The introduction of the Price Engine is especially easy for customers already using a SAP BW with Retail Cubes. Pre-configured online-access to the master and dynamic data in this database is available with the product. With it, the Price Engine always has the most current data.

Online-access to data of already used SAP BW

## System Environment Requirements

The Retail Simulation Engine runs on Windows-systems with .NET 2.0 on the server and client. The user authentification requires a Single-Sign-On in a Windows domain.

The connection to databases and data warehouses requires the availability of ADO-.NET data providers for the database being used. Direct access to the data warehouse should be available for larger amounts of data. The implementation of the Retail Simulation Engine through replication in a stand-alone database is however, also possible. Application data is filed in a separate database. Oracle 10g and Microsoft SQL Server 2005 are preferred here.

## Typical System Requirements

Low hardware requirements...

A Retail Simulation Engine-application server for 5 simultaneous optimization resp. terminal server users (equivalent to approx. 20 simultaneous users) and Oracle Workspace-database should fulfil the following performance requirements:

- 2 x Dual-core server processor (performance category Xeon 5130 / 2 GHz or higher)
- 4 GB main memory (+512 MB per parallel RDP-User)
- 120 GB hard disk space (Raid 1, if increased system stability is desired). The Dacos Price Engine does not put higher performance requirements on the disk performance; the performance of the hard disks primarily affects the workspace database aus.

A commercial desktop computer can be used as a client-computer; it should however be equipped with at least 512 MB RAM (assortment prognosis parallel with Excel: 1 GB) and have approximately 50 MB free memory on the hard disk.

# Customer-Specific Adjustments

Customer-specific adjustments possible

The automated integration of competitor prices collected and stored in a database is possible if these prices are available for the individual articles or article collections (with article number). The automatic provision of strategic price rules from marketing (the price intervals defined for your own private brands to brand-name products) can also be defined and implemented in a consultancy project.

In addition, it is also possible to extend the Tactical Price Optimization with reporting functions in order to bring optimization potential to the user's attention regularly. In principal, all of the key figures available in the data warehouse can be integrated into the Price Engine's information display.

# Literature

[1] Krafft, M., Mantrala, M.: Retailing in the 21st Century, Current and Future Trends, Springer Verlag, Heidelberg 2006

[2] Phillips, R.: Pricing and Revenue Optimization, Stanford Business Books, Stanford 2005

[3] Mitchell, M.: An Introduction to Genetic Algorithms. The MIT Press, Cambridge 1998.

[4] Data requirements Tactical Price Optimization.

© Content and design of this document are protected by copyright. The reproduction of this document, in full or in part, is permitted only with the express approval of the Dacos Software Ltd.

DACOS®
SOFTWARE

contact

Dacos Software GmbH
Science Park 2
D-66123 Saarbrücken
Fon +49 681-3946 71-0
Fax +49 681-3946 71-19
www.dacos.com