

NanoTrader

TradingSystems

Document Version 3.0

www.fipertec.com





Copyright

© Fipertec SARL. All rights reserved. No part of this document may be reproduced without the prior written permission of Fipertec SARL. The technology of NanoTrader is intellectual property of Fipertec SARL. This software and the accompanying files are sold "as is" and without warranties as to performance or any other warrants whether expressed or implied. In particular, Fipertec SARL is not liable for any financial disadvantages when following signals generated by NanoTrader. Any liability of Fipertec SARL will be limited exclusively to product replacement or refund of purchase price.

Contents

1	Intro	oduction	9
2	Nan	oTrader TradingSystems – An Overview	9
	2.1	Sentimentors – Indicators made alive	9
	2.2	Combining Sentimentors to a Study	11
	2.3	White Box Systems – Total Clarity	14
	2.4	Instant Backtesting	14
	2.5	Tradingsystems for Discretionary Trading – Decision Support	14
	2.6	Using Stops	15
	2.7	Optimization and Robustness	15
	2.8	Cascading Studies – Intermarket Analysis and Multiple Time Frames	16
	2.9	No Programming Required	17
	2.10	Going Live – Let NanoTrader Trade for You	17
	2.11	Enjoy NanoTrader	17
3	Disc	claimer Concerning Automatic Order Creation	17
4	Trac	ding Approaches	18
	4.1	Future Trading	18
	4.2	Performance Trading	18
	4.3	Trendsignals	19
	4.4	Evaluators	19
	4.5	Visualization of Trading Signals	19
	4.6	Visualization of Trendsignals	21
5	Dete	ermining the Sentiments	22
	5.1	Interpretation Scheme: Trigger Line	22
	5.2	Interpretation Scheme: Bands	25
	5.3	Interpretation Scheme: Two Zones	26
	5.4	Interpretation Scheme: Swing	26
	5.5	Interpretation Scheme: Support/Resistance	27
6	The	Meta Sentimentor	27
	6.1	Meta Sentiments as Weighted Average	28
	6.2	Free Definition of the Meta Sentiments	28
7	Thre	esholds of the Trading Approaches	33
	7.1	Thresholds for Future Trading and Performance Trading	33
	7.2	Thresholds for Trendsignals	34

8 Add	ling Sentimentors as Filters or Stops	. 34
8.1	Adding a Sentimentor as a "Filter"	. 34
8.2	Adding Stops to a study	. 36
8.3	Adding a Sentimentor as a Stop	. 36
9 Ana	alyzing a Trading Approach – The InfoBar	. 37
9.1	The Eval-Page	. 38
9.2	The Data Page	. 41
9.3	The Sentis Page	. 41
10 D	Defining a TradingSystem	. 42
10.1	Selecting the Trading Approach	. 42
10.2	The General Building Blocks	. 43
10.3	Defining the Parameter Ranges	. 43
10.4	Visual Feedback on Parameter Changes	. 45
10.5	Reset of a Parameter	. 45
10.6	Fixing a Parameter	. 45
10.7	Resetting all Parameters	. 46
10.8	Resetting all Parameters of the active Sentimentor	. 46
10.9	Exporting the Active Sentimentor	. 46
10.10	The Meta Sentimentor	. 46
10.11	Long / Short Positions	. 47
10.12	Displaying the Equity-window	. 47
10.13	Using the "Safety Net" for Automatic Trading	. 48
10.14	Creating a Report for the Trading Approach	. 50
11 E	valuator-Settings Dialog	. 51
11.1	Future Trading Evaluator	. 51
11.	1.1 Slippage and Transaction Costs	. 51
11.	1.2 The "Warm-Up Phase"	. 52
11.	1.3 Execution of Signals	. 52
11.2	Performance Trading Evaluator	. 56
11.3	2.1 General Settings	. 56
11.3	2.2 Money Management	. 56
11.3	The Trendsignal -Evaluator	. 58
12 E	valuation Time Periods	. 61
12.1	General Notes	. 61
12.2	Visualization of the Time Ranges	. 61

12.3	Specifying the Evaluation Time Range		
12.	3.1 Specifying the Control Time Range	62	
12.	3.2 Tail Time Range	63	
12.4	Switching to the Tail Time Range	63	
13 C	Optimization Goals and Constraints for Trading Strategies	64	
13.1	General Notes	64	
13.2	Optimization Goals	65	
13.3	Defining Constraints		
13.4	Optimizing with Embedded Sensitivity Analysis		
13.5	Optimization including the Control Period	67	
14 C	Optimization of a Tradingsystem		
14.1	General Notes on Optimization Techniques	67	
14.2	Optimizing a Single Parameter		
14.3	Optimizing a Single Sentimentor		
14.4	Optimizing the Complete Tradingsystem		
15 T	he Optimize Dialog	70	
15.1	Defining the Stop Criterion	71	
15.2	Animation	72	
15.3	Progress control		
15.4	Optimization History		
16 S	Sensitivity Analysis		
16.1	General Notes		
16.2	Sensitivity Analysis of a complete Study		
16.3	Sensitivity Analysis of a Sentimentor		
16.4	Sensitivity Analysis of a Parameter		
16.5	Trade Histogram		
17 S	Scripts for Automated Study Optimization		
17.1	Running a Script		
17.2	Editing a Script		
17.	2.1 Selecting Securities		
17.	2.2 Defining the Runtime per Security		
17.	2.3 Defining the Study Label and Standard Study		
17.3	Creating a "Template Study"		
17.4	17.4 Exporting MetaSentimentors		
17.5	Creating a New Script		

17.6	Cre	ating a New Script from an Existing Script	85		
17.7	Dele	eting a Script			
17.8	Sta	ting Scripts from the Command line			
17.9	Scre	eenerBar			
17.9	9.1	The ScreenerBar Columns			
17.9	9.2	Violations of Trading Constraints			
17.9	9.3	The ScreenerBar Context-Menu			
17.10	С	reating a Status-Report			
17.11	F	ormat of Script Files			
17.12	L	ogfiles			
18 N	lanua	ally defined Sentimentors			
18.1	Area	as of Application			
18.2	Edit	ing a Manual Sentimentor	91		
18.3	Cre	ating a New Manual Sentimentor			
18.4	Cre	ating a new Manual Sentimentor from an existing			
18.5	Dele	eting a Manual Sentimentor			
18.6	Add	ing a Manual Sentimentor to a Study			
18.7	18.7 Manual Sentimentors used as Filters				
18.8	The	"Flat" and "Block" Filters – Exiting at the end of the day .			
18.9	Cre	ating a Manual Sentimentor with an External Tool			
19 P	Param	neters of the Sentimentors			
19.1	Sen	timentors Based on Technical Indicators			
19.1	1.1	ADX			
19.1	1.2	Aroon			
19.1	1.3	ATR			
19. ⁻	1.4	Bollinger Bands			
19. ⁻	1.5	CCI – Channel Commodity Index	100		
19. ⁻	1.6	Channel Breakout	101		
19. ⁻	1.7	Crossing MA	102		
19. ⁻	1.8	Directional Ind. (+/-DI)	102		
19. ⁻	1.9	DMI – Dynamic Momentum Index	102		
19.1	1.10	DSS – Double Smoothed Stochastik	103		
19.1	1.11	Exponential Moving Average	104		
19. ⁻	1.12	Heikin Ashi	104		
19.1	1.13	KAMA - Kaufmann's Adaptive Moving Average	104		

19.1.14	Linear Regression	105
19.1.15	5 Local Highs & Lows	105
19.1.16	6 MACD	106
19.1.17	7 MACD-Histogram	106
19.1.18	3 Momentum	107
19.1.19	9 Moving Average	107
19.1.20	On-Balance-Volume	107
19.1.2 <i>′</i>	1 Orderbook	108
19.1.22	2 Parabolic SAR	108
19.1.23	B PFE – Polarized Fractal Efficiency	109
19.1.24	Pivot-Points	109
19.1.25	5 Point & Figure	110
19.1.26	8 Renko	111
19.1.27	7 RSI	112
19.1.28	3 RSI-smoothed	112
19.1.29	9 Slow Stochastic	113
19.1.30) Stochastic	113
19.1.3	1 Study	114
19.1.32	2 SuperTrend	114
19.1.33	3 Williams' Variable Accumulation Distribution (WVAD)	114
19.1.34	4 Williams %R	115
19.1.35	5 Volume	115
19.2 Ca	andle Stick	116
19.2.1	Visualization of Candle Stick patterns	116
19.2.2	Deactivation of the Trend Component	118
19.2.3	The Recognized Candle Stick Patterns	118
19.2.4	The Parameters of the Candle Stick Sentimentor	120
19.3 Su	ipport/Resistance-Sentimentor	121
19.3.1	The Computation of Support/Resistance-Lines	121
19.3.2	Using the Support/Resistance-Sentimentors in Constant Studies	122
19.3.3	The Parameters of the Support/Resistance-Sentimentor	122
19.4 Tr	endline Sentimentor	122
19.5 Ma	anually defined Sentimentors	123
20 Case	cading Studies	124

20.1	Ger	neral Notes	124
20.2	Usi	ng Cascading Studies	126
20.2	2.1	Embedding a Study	126
20.2	2.2	Embedding a Price Chart	127
20.2	2.3	Displaying an Embedded Price Chart in the MasterChart	129
20.3	Par	ameters of the Study-Sentimentor	130
20.4	Par	ameter "Sentiment Mapping"	130
20.4	4.1	Mapping Style "As Is"	130
20.4	4.2	Mapping Style "Map Thresholds"	130
20.4	4.3	Using an Embedded Study as a Filter	131
20.5	Par	ameter "Optimize Embedded Study"	132
20.6	Sor	ne Technical Details	132
20.7	Tim	escale Matching	132
20.8	Inh	erent Difficulty with Timestamps	133
21 V	Valkf	orward Analysis with the SystemTester	134
21.1	Ove	erview	134
21.2	Sta	rting the SystemTester	134
21.3	Cre	ation of Studies	136
21.4	Vis	ualization within the Study Creation Process	137
21.5	Vie	wing the Created Studies	138
21.6	Exe	ecuting the Signals of the Underlying Study	139
21.7	Per	formance Report	139
21.8	Sav	ring and Closing the SystemTest-dialog	139
21.9	Sor	ne Hints for Using the SystemTester	140
22 F	eedt	back	140

1 Introduction

This document concentrates specifically on the creation of trading systems, optimization, robustness and Walkforward simulation provided as an optional module of NanoTrader. You will also find the description of the built-in indicators in this document.

It is assumed that you are already familiar with the document **NanoTrader – Charting and Trading** which explains the basic usage of NanoTrader and specifically NanoTrader's concept of a TradeGuard – a specialized usage of a trading system that just handles the exits of positions.

To create a trading system with NanoTrader you will need a special permission. Please contact your broker or Fipertec if you want to obtain that permission.

2 NanoTrader TradingSystems – An Overview

2.1 Sentimentors – Indicators made alive

The basis for trading systems created with NanoTrader are so-called *Sentimentors*. A sentimentor is a building block that assigns a *sentiment* to each period of the chart. A *sentiment* is a value on a scale of 0 to 100. On this scale 0 represents the worst sentiment and 100 the best. For example, the "Crossing Moving Average" sentimentor would assign a 100 on upward crossings and a 0 on downward crossings.

By this simple and intuitive approach each indicator becomes "alive". It is not just a drawing on the screen that the trader has to interpret over and over again. Instead, it emits a sentiment at each period thus giving an evaluation of the market from the indicator's perspective. If a sentiment for a given period rises above or falls below a threshold that sentiment is converted into a *signal*. Signals of a sentimentor will be visualized in their charting window and optionally in the main chart as well:



The sentiments being emitted are easily defined through dialogs – there is no programming required!

Interpretation for a Trigger Line		×
1.	4.	
Event:	Sentiment Series starting at event period, e.g. 100;90	
1. Crossing above Trigger:	100;	
2. Staying above Trigger:	65	
3. Crosing below Trigger:	0;	
4. Staying below Trigger:	35	
Standard -	Select	
Remove	Save	
ОК	Cancel	

The following graphic shows the principle of how a sentimentor creates a sentiment series:





Throughout this document we will use the terms indicator and sentimentor as synonyms.

Also we will use the terms *trading system* and *study* as synonyms.

2.2 Combining Sentimentors to a Study

NanoTrader allows you to combine any indicators of your choice in a study. Naturally different indicators will have different indications at certain point in time, some may tell you "go long", some will say "don't know", others will say "go short" – we call this "the technical trader's dilemma". As an example consider a typical setup of study consisting of Pivot Lines, a Moving Average, a MACD and a RSI:



A standard charting application would let the trader alone with converting the drawings into some meaningful interpretation. Obviously this consumes an enormous amount of energy by the trader and the outcome of the ad-hoc analysis will surely vary with the trader's current mental and monetary condition.

NanoTrader allows to standardize and automate the evaluation by condensing the individual sentiments emitted by each indicator to an overall sentiment. This is achieved by the so-called *MetaSentimentor*:



This elegant usage of sentiments allows you to include anything into a trading system: indicators, formation analysis, fundamental analysis, Fibonacci retracements, candlestick formations, etc. In addition, fuzzy information and intuitions can also be combined and analyzed in a standardized form.

The following graphic shows how the sentiment series of different sentimentors are combined by the MetaSentimentor:





2.3 White Box Systems – Total Clarity

As opposed to "black box" systems like, say, neural networks, NanoTrader is a 100% white box system – you always know *why* a signal is there. The trader defines the rules, not some built-in magic.

2.4 Instant Backtesting

NanoTrader's *Instant Backtesting* will display all trading signals and statistics continuously as you change your trading system parameters and as live data arrives at your system – there is no need for specific procedures to evaluate the system.

2.5 Tradingsystems for Discretionary Trading – Decision Support

A trading system in NanoTrader could be designed for supporting your discretionary activities in various ways. For example sentimentors might be used as filters coloring the chart in green or red whenever they find an upward or downward trend. You as a trader then wait for the best moment according to your personal style for actually entering a position.





2.6 Using Stops

The MetaSentimentor gives an evaluation for each period according to the sentimentors that were added to the study. Depending on the thresholds defined in the study the MetaSentimentor can open, close, keep, or reverse a position.

Besides the MetaSentimentor which is continually reflecting the overall market, you might want to have additional components for closing a position, e.g.:

- secure a profit by usage one or multiple trailing stops or a profit targets
- close the position at certain time before the end of the trading session
- close the position as soon as a trendline is crossed, etc.

2.7 Optimization and Robustness

Once you combined the components of a study you surely want to optimize it – just to make sure you get the best out of your trading idea. Don't even think you could do it manually – even the simplest trading idea requires a couple of parameters: Assume a system consisting of a Moving Average (one parameter), an MACD (three parameters), and a trailing stop (one parameter). Let each parameter have just 50 feasible settings and you will end up with 312.500.000 possible combinations.

Please, do *not* buy into this myth stating "My trading system is robust, because it has no parameters!". There cannot be a trading system without parameters. A parameter could be *hidden* by defining it as being constant, but it obviously still influences the outcome of the calculations and hence the profit and robustness.

NanoTrader is equipped with the fastest and most flexible optimization engine currently available on the market. This will give you the power to test and polish your trading ideas in-depth.

Because of its speed NanoTrader not only allows for an unlimited number of parameters to be optimized, but also includes built-in functionality for verifying the robustness of your system.



The two most important things you as the designer of a trading system have to keep in mind are:

- The selected sentimentors and their configuration need to represent a precise trading idea. If you just randomly combine some sentimentors you will have no basis for creating trust in the system no matter how good the statistics look.
- Each parameter has to be assigned a range in which the optimizer can vary the parameter. That range needs to be defined so that the trading idea is preserved no matter what actual value the parameter takes.

With this understanding the NanoTrader optimizer simply helps you to detect the best parameter combination for the trading idea *that you defined by selecting the sentimentors and the parameter ranges.* Thus, NanoTrader gives you an enormous flexibility in what you could incorporate into a study, but it is always you who is driving the creation process.

2.8 Cascading Studies – Intermarket Analysis and Multiple Time Frames

NanoTrader is unique in its capability in including the result of one study as the input for another, e.g., use a study on the Dow as a filter for trading the S&P, or use an analysis of the Gold future as an input for a study of a mining company.

You might as well include a study based on daily data into a 10-minute chart so you will always trade in sync with the broader time range.

2.9 No Programming Required

Trading systems in NanoTrader can be created without any programming. You just specify the sentiments of the sentimentors you apply in your study through straightforward dialogs, and that's all it takes.

Additional building blocks *could* be programmed with NanoTrader-Express or they could be retrieved from NanoTrader's community to enrich the environment. Each Express sentimentor acts exactly as an built-in sentimentor, i.e., it can be combined with other sentimentors, configured, and optimized.

2.10 Going Live – Let NanoTrader Trade for You

Once you have tested and fine tuned your study you could let it trade for you. It requires just a click:



2.11 Enjoy NanoTrader

NanoTrader has a rich functionality to explore. In addition to reading this documentation you should view some of the videos presented in the Help Center – this will give you an excellent start for the usage of NanoTrader.

For learning more about creating and testing trading systems please see the list of Fipertec seminars on this topic as well as online lectures given at <u>www.fipertec.com</u> and www.fipertec.de.

Fipertec is greatly interested in receiving feedback from you and to add functionality aiding you in your daily trading processes. Please do not hesitate to submit your feedback to <u>info@fipertec.com</u>.

3 Disclaimer Concerning Automatic Order Creation

Automatic order creation requires a complex infrastructure where all components have to work accurately (e.g., PC stability, internet, broker, trading platform, data provider, study configuration). Therefore, Fipertec rejects any liability for damage resulting from using NanoTrader in the trading process.

In particular when using the automatic order creation provided by NanoTrader the trader is advised to continually observe the trading activities initiated by NanoTrader!

4 Trading Approaches

A trading approach defines how to convert a given sentiment series into signals. The trading approach is selected in the DesignerBar:



4.1 Future Trading

The given signals form a series of buy /sell transactions. Transaction costs and slippage influence the performance. The profit of a trade is computed based on the point value of the future and the entry and exit prices. The number of contracts per transaction is fixed.

4.2 Performance Trading

This approach is used for trading stock. The given signals form a series of buy/sell transactions. Transaction costs and slippage influence the performance. The profit of a trade is computed based on the entry and exit prices.

A special variant is the "Percentage Trading" where the profit is measured in percentages. The number of stocks to be bought is controlled through Money Management.

4.3 Trendsignals

Trendsignals are generated independently of each other, i.e. long and short signals are given without a reference to the preceding signals. Moreover, transaction costs and budgets are not taken into account.

4.4 Evaluators

Associated with each Trading Approach is a so-called *Evaluator*. An Evaluator holds a number of settings that define precisely how to convert signals into trading actions, how to measure the slippage etc. The Evaluators are described in a dedicated <u>section</u>.

4.5 Visualization of Trading Signals

The signals generated by the Future Trading and Performance Trading approaches are visualized as follows:

	solid green triangle	long (buy/buy calls/buy contracts)
•	small solid green triangle	long-signal to be executed in the next period
		This signal type is displayed only if the Signal Execution for Sentiment Enter-signals is set to "Confirmation price next bar", "Limit price next bar" or "Open next bar" in the Evaluator. Moreover, the field draw setup signals of the Options-dialog must be activated.
	solid green triangle with bar	close long triggered by a sentiment signal
		In case the triangle is filled with gray, the signal was generated by a Stop.
		If it is filled with yellow, the profit target has been reached.
•	solid red triangle	short (sell short/buy puts/sell contracts)
•	small solid red triangle	short signal to be executed in the next period
		This signal type is displayed only if the Signal Execution for Sentiment Enter-signals is set to "Confirmation price next bar", "Limit price next bar" or "Open next bar" in the Evaluator. Moreover, the field draw setup signals of the Options-dialog must be activated.
-	solid red triangle with bar	close short triggered by a sentiment

	signal
■	In case the triangle is filled with gray, the signal was generated by a Stop.
1	If it is filled with yellow, the profit target has been reached

Depending on the chosen Signal Execution policies a bar may create several signals. If the mouse points to a bar, all signals generated within this bar are explained in a popup window.



The order in which the signals are explained corresponds to the order the signals have been executed by NanoTrader.

A red and a green triangle displayed at the same period (e.g., \mathbf{X} or \mathbf{X}), indicate a reversal of the position.



Note that the last signal is displayed enlarged so you can easily spot it. The magnitude of the enlargement can be defined through Extras|Options.

For the performance analysis an open position is handled as if it were closed at the last bar of the evaluation period.

4.6 Visualization of Trendsignals

The following scheme is used to visualize trendsignals:

•	solid green triangle	correct long signal
	green triangle	wrong long signal
•	solid red triangle	correct short signal
~	red triangle	wrong short signal
•	green triangle filled with blue	A buy signal that is so near to the present that its correctness cannot be evaluated yet (open buy signal).
•	red triangle filled with blue	A sell signal that is so near to the present that its correctness cannot be evaluated yet (open sell signal).

The following screenshot shows trendsignals in a MasterChart



The green areas show the target zones that have to be exceeded, whereas the red areas show the stop zones that may not be exceeded.

5 Determining the Sentiments

The conversion of an indicator into a sentiment series is called *interpretation*. A sentimentor is always a combination of an indicator (in its broadest sense) and an interpretaion. NanoTrader allows for nearly all sentimentors to configure their respective interpretation using simple to use dialogs.

In most of the cases the interpretation is performed using one of five standardized interpretation schemes. These schemes are explained in detail in the following sections.

The configuration dialog associated with a given sentimentor is opened by either doubleclicking the sentimentor in the DesignerBar or by first activating the sentimentor and then clicking the $\frac{1}{\sqrt{2}}$ icon in the toolbar of the DesignerBar or by rightclicking the sentimentor and then choosing Edit Interpretation from the context menu.



The principle way of working with an interpretation configuration dialog is described using the dialog for the interpretation scheme "Trigger Line". The other dialogs work in exactly the same way. The provided screenshots show the default setting of the schemes used by NanoTrader.

5.1 Interpretation Scheme: Trigger Line

The dialog for configuring the interpretation scheme Trigger Line looks as follows:

Interpretation for a Trigger Line	×
1.	4.
Event:	Sentiment Series starting at event period, e.g. 100;90
1. Crossing above Trigger:	100;
2. Staying above Trigger:	65
3. Crosing below Trigger:	0;
4. Staying below Trigger:	35
Template	
Ψ	Select
Remove	Save
ОК	Cancel

All dialogs for configuring an interpretation scheme are divided into three parts. The upper part contains a graphic explaining the scheme. The middle part consists of a number of fields for defining the sentiments, and the lower part allows provides functionality to work with *templates*.

The interpretation of an indicator is based on so-called *events*, e.g., the crossing of two lines or the entering of a curve into a zone. At the Trigger Line interpretation scheme these events are caused by two lines, as known from the classical "Crossing Moving Averages" sentimentor – here the slow moving average plays the role of the trigger.

The graphic of the configuration dialog shows four different events. The dotted line represents the trigger line.

Event 1 occurs whenever the trigger line is crossed from below. The sentiment used to signify this event is defined in the corresponding input field:

1. Crossing above Trigger: 100;

Analogously the sentiments can be defined for the events 2, 3, and 4. Recall that a sentiment is a value between 0 and 100.

The interpretation distinguishes between *primary* and *secondary* events. In the Trigger Line scheme, the crossing events (events 1 and 3) are the primary events. The sentiment input fields for primary events are larger as those for secondary events. For primary events, NanoTrader allows to specify the sentiment for the bar where the event occurred and for an arbitrary number of subsequent bars. This can be achieved by specifying several sentiments separated by semicolons:

1. Crossing above Trigger:	100;100;100
----------------------------	-------------

Using this specification the good sentiment of 100 would carry over to the two bars following the "event bar". This technique is especially useful when combining several sentimentors that have to show similar sentiments at "roughly" the same bar.

A fading of the sentiment is also possible, as in:

1. Crossing above Trigger: [100); 95; 90; 85; 80; 75

Such a sequence of sentiments can be stopped if a new primary event occurs, e.g., if two periods after an upward crossing of the trigger line a downward crossing occurs. Secondary events cannot stop a sequence originated by a primary event.

Working with Templates

Usually an interpretation configuration that has been defined with a certain idea in mind should be reusable in other sentimentors that apply the same interpretation scheme. For this reason NanoTrader allows to save a configuration setting as a *template*.

Interpretation for a Trigger Line	×
1.	4.
Event:	Sentiment Series starting at event period, e.g. 100;90
1. Crossing above Trigger:	100;95;90;85;80;75;
2. Staying above Trigger:	65
3. Crosing below Trigger:	0;5;10;15;20;25;
4. Staying below Trigger:	35
Template	
Fade Out 🗸	Select
Remove	Save
ОК	Cancel

To save the current setting as a template, first specify a symbolic name and then click on the Save button. To load a template, select it from the list of available templates and click Select.



Interpretation for a Trigger Line	×
1. 2. 3	4.
Event:	Sentiment Series starting at event period, e.g. 100;90
1. Crossing above Trigger:	100;95;90;85;80;75;
2. Staying above Trigger:	65
3. Crosing below Trigger:	0;5;10;15;20;25;
4. Staying below Trigger:	35
Template	
Fade Out 🔹	Select
Continuous	Save
Fade Out	
longSentiments	Canael
Standard	

5.2 Interpretation Scheme: Bands

Interpretation for Two Bands	×
1 2 5 6	3.
Event:	Sentiment Series starting at event period, e.g. 100;90
1. Crossing above Upper Band:	100;
2. Staying above Upper Band:	75
3. Crosing below Upper Band:	35;
4. Staying between Bands:	50
5. Crossing below Lower Band:	0;
6. Staying below Lower Band:	25
7. Crossing above Lower Band:	65;
Template	
*	Select
Remove	Save
ОК	Cancel



5.3 Interpretation Scheme: Two Zones

Interpretation for Two Zones	×
1.	3.
2.	4
	. 7.
Event:	Sentiment Series starting at event period, e.g. 100;90
1. Entering Upper Zone:	50;
2. Staying in Upper Zone:	50
3. Leaving Upper Zone:	0;
4. Staying between Zones:	50
5. Entering Lower Zone:	50;
6.Staying in Lower Zone:	50
7. Leaving Lower Zone:	100;
Template	
-	Select
Remove	Save
ОК	Cancel

The interpretation scheme Zones uses two parameters "Threshold Up" and "Threshold Down" that are defined by the sentimentors using this scheme.

5.4 Interpretation Scheme: Swing

Interpretation for Swings	×
2.	3. 4.
Event:	Sentiment Series starting at event period, e.g. 100;90
1. End of Downward Swing:	100;
2. Upward Movement:	50
3. End of Upward Swing:	0;
4. Downward Movement:	50
Template	
·	Select
Remove	Save
ОК	Cancel

The interpretation scheme Swing relies on two parameters: Span Left: examination range "before" an extreme point Span Right: examination range "after" an extreme point The end of a "Downward Swing" (Event 1) is given, if the examined curve falls "Span Left" bars and then rises "Span Right" bars.

In "Upward Movement" occurs if the value of the current bar is greater than that of the previous bar.

"End of Upward Swing" and "Downward Movement" are defined analogously.

5.5 Interpretation Scheme: Support/Resistance

Interpretation for Support/Resistance Line		
4. 5.	6.	
1. 2.	7.	
Event:	Sentiment Series starting at event period, e.g. 100;90	
1. Entering Resistance Zone:	25;	
2. Staying in Resistance Zone:	25	
3. Crossing above Line:	100;	
4. Staying in Support Zone:	75	
5. Leaving Support Zone:	75;	
6. Entering Support Zone:	75;	
7. Crossing below Line:	0;	
8. Leaving Resistance Zone:	50;	
Template		
T	Select	
Remove	Save	
ОК	Cancel	

6 The Meta Sentimentor

The role of the Meta Sentimentor is the computation of the overall sentiment based on the individual sentimentors constituting the study. To do this NanoTrader provides two different mechanisms:

- the computation of a weighted average of the sentiments
- the free definition of the Meta Sentiments based on simple logical expressions (*if then conditions*)

6.1 Meta Sentiments as Weighted Average

This is the default mode when adding the MetaSentimentor to a study. NanoTrader automatically creates one parameter per sentimentor that defines the weight the sentimentor has in the weighted average:



In the example above the systems consists of a Crossing MA sentimentor, a RSI sentimentor, and the MetaSentimentor. The Crossing MA is assigned a weight of 3, the RSI has a weight of 1. Hence, the impact of the Crossing MA is three times as strong as that of the RSI.

Computation:

First of all the weighted sum of the sentiment series of the sentimentors is computed. The weights to be used are defined by the weight parameters. Then a moving average (MA) on the weighted sum is computed.

Parameter:

Smoothness-Span: speed of the MA

per sentimentor: the weight to be used for computing the weighted sum

Interpretation:

The moving average is the resulting sentiment series that will be used for generating signals by the trading approaches. Whenever the Meta Sentiment of a period rises above or falls below a certain threshold (see Section <u>Thresholds</u> of the <u>Trading Approaches</u>) a signal is generated.

6.2 Free Definition of the Meta Sentiments

Sometimes it is desirable to define the Meta Sentiment based on a number of "if then" conditions, e.g.

If the CCI indicates an upward trend and there is an upward crossing in the "Crossing Moving Average" sentimentor then the Meta Sentiment is 100.

Conditions of this kind are commonly used in the "classical" approaches for defining trading systems.

NanoTrader allows to define such rules by accessing the sentiments of sentimentors used in the study.

Starting the Condition Editor

The Condition Editor is started by double clicking the Meta Sentimentor in the DesignerBar:



The Editor looks as follows:



Format of a Condition

A condition, also called a *logical expression*, has the following general format:

<condition> : sentiment ;

To state a *condition*, the following elements can be used:

- The name of a Sentimentor used in the study. Such a name represents the sentiment value of the respective sentimentor at the currently processed bar.
 Some names of Sentimentors contain spaces or special characters for referring such a Sentimentor, these characters have to be omitted, e.g. "Crossing MA" is referenced by CrossingMA and "Directional Ind. (+/-DI)" by "DirectionalIndDi".
 The names are not case sensitive, i.e., you may use capital or lower characters at your wish.
- The relational operators
 - > greater than
 - < less than
 - >= greater than or equal
 - <= less than or equal
- The logical operators AND and OR.
- The numbers 0 100 as the sentiment values.
- Parenthesis to group expressions

Here is an example for a complete condition:

(CCI > 70) and (CrossingMA = 100) : 100;

The condition (CCI > 70) and (CrossingMA = 100) refers the sentiments of the CCI Sentimentor and "Crossing MA" Sentimentor. The resulting Meta Sentiment, in case the condition is met, is defined as 100. This sentiment value is separated from the condition by a colon. Finally, a condition is ended by a semicolon.

An arbitrary number of conditions can be defined:



E Interpretation through Conditions	×
Conditions have the following format: CONDITION : SENTIMENT; SENTIMENT;; Examples for Conditions: (CCI >= 90) and ((CrossingMA = 100) or (RSI = 100) : 100; 100; 100; (CCI = 0) or (CrossingMA = 0) : 0;	
Delete Text	
(CCI > 70) and (CrossingMA = 100): 100; (CCI < 30) and (CrossingMA = 0) : 0;	▶
4	r
Template Select Remove Save	
OK Cancel	:

Evaluation of the Conditions

NanoTrader computes the Meta Sentiments beginning with the first bar (the "leftmost" bar) and then moving bar by bar to the final bar.

For each bar the defined conditions are checked in the order of their definition. In case a condition is met then the sentiment defined for this condition becomes the Meta Sentiment and the subsequent conditions are ignored.

In case no condition is met for a given bar then the Meta Sentiment for this bar is set to 50, i.e. neutral.

Assigning Multiple Sentiments

If a condition is met that represents a *strong* signal, then sometimes it is interesting to assign a Meta Sentiment not only for the given bar where the signal occurred, but also to some following periods. Thus, a signal can be converted into a trading action even some bars after its initial occurrence in case the general condition, like filters or confirmation prices, are also met.

Using the notation

<condition> : sentiment ; sentiment ; ... sentiment ;

the Meta Sentiment is assigned to the actual period and the subsequent periods.



Example:

(CCI > 70) and (CrossingMA = 100) : 100; 100; 100;

In this example, the signal is "valid" for three periods.

Comments

The Condition Editor interprets all characters following an double slash // until the end of the line as a comment. Comments can be used to structure the conditions and to make them more readable.

Interpretation through Conditions		×
Conditions have the following format: CONDITION : SENTIMENT; SENTIMENT;;		
Examples for Conditions:		
(CCI >= 90) and ((CrossingMA = 100) or (RSI = 100) : 100; 100; 100; (CCI = 0) or (CrossingMA = 0) : 0;);	
Γ	Delete Text	
<pre>//Long Position (CCI > 70) and (CrossingMA = 100): 100;</pre>		*
//Short Position (CCI < 30) and (CrossingMA = 0) : 0;		
4	Þ	~
Tamalata		
- Select		
Remove Save		

Using multiple Instances of one Sentimentor Type

If a study uses multiple sentimentors of the same type, e.g., two RSI sentimentors, then referring to these sentimentors requires an indexing:

(RSI1 > 70) and (RSI2 > 50) : 100;

The first (up most) RSI Sentimentor in the Designer table is denoted RSI1, the second RSI2, and so forth.

Templates

Once a set of conditions has been specified it can be saved as a *template* for reusage in other studies. To do this, enter a symbolic name for the template and then click on the Save button:



E Interpretation through Conditions	×
Conditions have the following format: CONDITION : SENTIMENT; SENTIMENT;; Examples for Conditions: (CCI >= 90) and ((CrossingMA = 100) or (RSI = 100) : 100; 100;100; (CCI = 0) or (CrossingMA = 0) : 0;	
Delete Text	
<pre>//Long Position (CCI > 70) and (CrossingMA = 100): 100; //Short Position (CCI < 30) and (CrossingMA = 0) : 0;</pre>	
	r
Template TrendConfirmation Remove Save OK Cancel	

Switching back to the Weighted Average

To change back from the conditions to the Weighted Average scheme, simply delete all conditions (probably after saving them as a template). This can be done most easily by clicking the Delete Text button.

When quitting the dialog by clicking OK NanoTrader will change to the Weighted Average computation.

7 Thresholds of the Trading Approaches

A trading approach defines how a sentiment series is used to generate signals. The generation scheme is driven by parameters that are also considered by the optimization. These parameters are shown in the entry "Trading" which appears always as the first entry of the Indicators section of the DesignerBar as soon as the MetaSentimentor is added to the study:

Design	erBar	
۰ 🚯	🖌 🗙 🗱 🖉	s≁ \$+ ₩ 🗉 🖬
🕘 Ind	licators	*
	Trading	76, 24, 40, 60
	Threshold Long	76
	Threshold Short	24
	Close Long	40
	Close Short	60

7.1 Thresholds for Future Trading and Performance Trading

The Trading approach relies on four parameters serving as thresholds.

Threshold Long: If a sentiment value hits or exceeds this threshold, a long signal is generated.

Threshold Short: If a sentiment value hits or falls below this threshold, a short signal is generated.

Close Long: A long position is closed in case a sentiment value hits or falls below this threshold.

Close Short: A short position is closed in case a sentiment value hits or exceeds this threshold.

7.2 Thresholds for Trendsignals

The *Trendsignals* approach relies on two parameters serving as thresholds.

Threshold Long: If a sentiment value hits or exceeds this threshold, a long signal is generated.

Threshold Short: If a sentiment value hits or falls below this threshold, a short signal is generated.

8 Adding Sentimentors as Filters or Stops

8.1 Adding a Sentimentor as a "Filter"

Quite often one wants to exclude certain trades if the overall market conditions do not meet certain requirements. E.g., many traders avoid long positions in long-term downtrend markets even if there is a good chance for a small rally. NanoTrader supports filters within studies to express this kind of restrictions. When using filters, an entry signal of the MetaSentimentor is executed only if it passes all filters, i.e., for a long entry signal, all filters need to show a sentiment of at least 65, and vice versa for short entry signals all filters must have a value of at most 35.

Click on the low toolbar button of the DesignerBar's toolbar to open the Add Sentimentor dialog, select a sentimentor, an click on the Filter button:



Add Sentimentor Type to search Indicators ADX ADX ATR Bollinger Bands Candle Stick CCI Channel Breakout Crossing MA Directional Ind. (+/-DI) DSS Exp. Moving Average Dyn.Momentum Idx Heikin Ashi High-Low Channel Kaufmann's AMA		X ↑ Crossing MA Volume Viewer Exp. Moving Average Time Price Opportunities Key Price Levels Volume Profile LiveStatistics Express\PrevDayClose Bollinger Bands Kaufmann's AMA Meta Sentimentor Express\EMA Trailing Stop Trailing Stop EoP MACD	
 Display in MasterChart Display as subwindow in MasterChart Insert as: Sentimentor Filter 	Willow	top Tactic Close	

Filters based on "Manual Sentimentors" can also be used to rule out trades at certain points in time, e.g., before the expiration date of options or before quarterly reports.

The regions where all filters of a study show a long or short sentiment can be visualized in the chart. This visualization allows to check at a glance at what points in time positions could be entered at all.



NanoTrader allows to add as many filters to a study as you like. The DesignerBar lists all filters in a special section:





8.2 Adding Stops to a study

Successful trading strategies require both excellent entry signals *and* excellent exit signals. Therefore, NanoTrader supports a variety of stop techniques in addition to exiting a position via the MetaSentimentor:

- **price based stops** (e.g. trailing stop) exit a position if a specifically computed price level is reached
- time based stops (e.g. End-of-Day stop) exit a position at a certain time
- **sentiment based stops** exit a position if the sentiment contradicts the current position. Each sentimentor can be used as such a "Stop-sentimentor. Each Stop-sentimentor is evaluated independently from each other and from the MetaSentimentor.

Price based stops and time based stops are discussed in the accompanying documentation *NanoTrader – Charting and Trading,* so we will concentrate here on sentiment based stops.

8.3 Adding a Sentimentor as a Stop

To add a sentimentor as a stop, first select the sentimentor and then add it by clicking on the Stop-button:


Type to search , 		X ↑ Crossing MA Volume Viewer Exp. Moving Average Time Price Opportunities Key Price Levels Volume Profile LiveStatistics Express\PrevDayClose Bollinger Bands Kaufmann's AMA Meta Sentimentor Express\EMA Trailing Stop Trailing Stop EoP MACD	•
 Display in MasterChart Display as subwindow in MasterChart w Insert as: Sentimentor Filter 	indow Stop	Tactic Close	

A stop-sentimentor closes a long position if it shows a sentiment of 0. A short position is closed if it shows a sentiment of 100. The stop leading to exiting a position is shown in a popup window when pointing to the exiting signal:



In the example given above, a Parabolic stop is used in conjunction with a RSI used as stop sentimentor. The popup window shows that the RSI is responsible for the exit signal.

Because stop sentimentors are based on sentiments, not on prices, they are always evaluated at the end of a period, not within a period.

9 Analyzing a Trading Approach – The InfoBar

To analyze the quality of a given sentimentor parameter setting NanoTrader continually produces an evaluation report being displayed in the InfoBar. The *Instant Backtesting* of NanoTrader makes sure that the report and signals are updated automatically as soon as a parameter is changed, a sentimentor is added or removed or new data arrives.

The InfoBar displays the reports independently for the optimization/control/tail periods:



9.1 The Eval-Page

The Eval-Page of the InfoBar displays the current evaluation of the trading system. The shown information varies with the different approaches.



tem	Optimization Per	Control Per.	Tail
Total net profit:	432.28	20.50	37.90
Total # of trades:	42	8	4
Winning trades:	42	7	4
Losing trades:	0	1	0
Percent profitable:	100.00%	87.50%	100.00%
Profit factor:	n/a	1.45	n/a
Avg win/avg loss:	n/a	0.21	n/a
Avg trade (win & loss):	10.29	2.56	9.47
Percent in the market:	40.62%	72.26%	34.67%
RegCoeff*100/StdDev E	0.0000	0.0000	0.0000
Gross profit:	432.28	65.80	37.90
Gross loss:	-0.00	45.30	-0.00
Largest winning trade:	64.30	10.00	10.00
Avg winning trade:	10.29	9.40	9.47
Avg # bars in winners:	21.24	42.00	19.50
Largest losing trade:	n/a	45.30	n/a
Avg losing trade:	n/a	45.30	n/a
Avg # bars in losers:	n/a	102.00	n/a
Max consecutive winn	42	7	4
Max consecutive losers:	0	1	0
Std.Dev. all trades:	8.60	19.36	1.05
Std.Dev. winning trades:	8.60	1.02	1.05
Std.Dev. losing trades:	0.00	0.00	0.00
Max # shares/contracts:	1	1	1
Max drawdown:	0.00	45.30	0.00
Commission paid:	0.00	0.00	0.00
Expectancy:	n/a	0.0566	n/a
Expectancy Score:	n/a	0.0008	n/a
Happiness Factor:	1000006.75	0.30	1000003.81
Performance/Drawdow	432.28	0.45	37.90
Expectation:	10.29	2.56	9.47
Evaluation start:	02.06.21 Wed 0	15.06.21 T	17.06.21 T
Evaluation stop:	15.06.21 Tue 19	17.06.21 T	18.06.21 F

The report consists of the following entries.

Name	Definition
total net profit	gross profit less the losses, commission, and slippage.
performance	performance with respect to the initial account size. (Displayed only if the strategy relies on an initial account size.)
total # of trades	total number of trades
winning trades	number of trades with a remaining profit after subtracting commission and slippage
losing trades	number of trades with ending in a loss (commission and slippage included)



percent profitable	the ratio of winning trades to losing trades
gross profit	sum of the profit of the winning trades (commission and slippage included)
gross loss	sum of the losses of the losing trades (commission and slippage included)
avg win/avg loss	ratio of the average winning trade to the average losing trade (gross profit / winning trades) / (gross loss / losing trades)
profit factor	gross profit / gross loss A "profit factor" of 3 would indicate a profit of \$3 per \$1 loss.
Avg trade (win & loss)	(gross profit + gross loss) / total # of trades
Largest winning trade	the profit of the largest winning trade (displayed as an absolute value and in percent of the total net profit)."
avg winning trade	average winning trade gross profit / winning trades
avg # bars in winners	average duration of a winning trade (currently, a <i>bar</i> is always one day)
largest losing trade	the loss of the largest losing trade
avg losing trade	average losing trade gross loss / losing trades
avg # bars in losers	average duration of a losing trade (currently, a <i>bar</i> is always one day)
max consecutive winners	greatest number of consecutive winning trades
max consecutive losers	greatest number of consecutive losing trades
Std. Dev. all trades	standard deviation of the net profit of all trades
Std. Dev. winning trades	standard deviation of the net profit of the winning trades
Std. Dev. losing trades	standard deviation of the net profit of the losing trades
percent in the market	ratio between the time an open position was kept and the total evaluation time span.
max # shares/contracts	size of the largest position held
max drawdown	maximal loss with respect to the highest preceding account size Example: Suppose after five trades the account size reached a maximum value of \$10.000, and the following trades result in an account size of \$2.000.



	At this point the drawdown is 80%.
commission paid	sum total of paid commission
included slippage	sum total of included slippage
Expectancy Score	[(PW x AW- PL x AL) / AL] * Opportunity with: PW = probability of winning AW = average winning trade PL = probability of losing AL = average losing trade Opportunity = # Trades / # Periods in the evaluation period
Happiness Factor	[Performance * PW * (min(20, ProfitFactor) + min(20, AW / AL))] / [max. drawdown + max. losing trade + max winning trade]
Expectation	PW x AW - PL x AL with: PW = probability of winning AW = average winning trade PL = probability of losing AL = average losing trade

9.2 The Data Page

The Data-Page displays information corresponding to the date the mouse is indicating.

The Average Range and Average True Range are computed for the currently selected evaluation period as selected in the DesignerBar.

The Range as used for the computation of the Average Range is the quotient of the high and low of the period expressed in percent. The True Range enlarges the trading range by taking the close price of the previous period into account. It is defined as the maximum value of the subtraction

- current period's high minus current period's low (the Range)
- current period's high minus previous period's close
- previous period's close minus current period's low

9.3 The Sentis Page

A major characteristic of NanoTrader is the fact that the reason why a signal has been generated can be verified by the user. This results from the fact that a signal is generated exclusively because of the sentiment values generated by the applied sentimentors. In order to make the signal generation as transparent as possible the Sentis-Page displays the respective sentiment values of the sentimentors applied in the current study for the date the mouse is indicating.

InfoLeiste 🗸 🕂 🗶						
Sentime	entor		Sentim	gewichtet	Wer	
Filter:C	CI (20, 1	100, -100)	100.00		165.26	
Meta S	entimen	tor (1, 2, 1)	76.67	MAspan=1	76.7	
Crossir	ng MA (30, 100)	65.00	43.33	1568	
Line Trendline (0.1, 0.1, 0, 0)			100.00	33.33	0.00	
•					•	
Daten	Sentis	Eval				

The Meta Sentimentor is built by computing the weighted sum of the sentimentors. Therefore, the column Weighted shows the weighted contribution of each sentimentor to the Meta Sentimentor for the date the mouse is indicating. This gives an immediate understanding of the generated signals.

In case the Meta Sentimentor is computed without applying a moving average, the sum of the Weighted column is exactly the Meta Sentiment value at the given date. In the example given above "MAspan = 1" denotes the span of the Moving Average is 1, i.e., no moving average is computed.

The last column Value displays the "most important" value of each sentimentor at the given point in time, e.g., the CCI has a value of 165.26.

10 Defining a TradingSystem

10.1 Selecting the Trading Approach

As discussed in section <u>Trading Approaches</u> NanoTrader distinguishes between three Trading Approaches used to evaluate a given trading system:

- FutureTrading: The performance of one trade is measured in the absolute difference of entry and exit prices multiplied with the value per point.
- Performance Trading The performance is measured in the absolute difference of entry and exit prices or in percentages. This approach is usually used for stocks.
- TrendSignals
 A (historical) specialty of NanoTrader that rates signals as correct or wrong. This approach cannot be used for automated trading.

The approach to be used for the trading system is selected in the DesignerBar:



DesignerBar 🛛					
6	- 🖌 🗙 😵	📇 🛹 \$+ 🐜 🔳 🖬			
	Trading				
	Account	DA200616			
	Order Volume	1			
	Automation state	Deactivated			
	Target Ordertype	Limit			
	Stop Ordertype	Stop			
	Brackets				
	Unit for Brackets etc.	Ticks			
	ATR Span	20			
	Multiple Targets	No			
	Multiple Stops	No			
	Information				
	HotKey				
⊳	Tactic Buttons				
⊳	Filters				
⊳	Indicators				
	Tradingsystem Settin	igs			
	Trading Approach	Future Trading			
	Equity Chart	Performance Trading			
	Display backtes	Future Trading			
	MetaSentiment	Trendsignals 😽			

10.2 The General Building Blocks

All building blocks for creating a trading system are assembled in the DesignerBar. A trading system consists of the following components:

- A (small) selection of sentimentors.
- The MetaSentimentor to summarize the sentimentors.
- Optionally Filters that rule out certain time periods, no-trend periods, etc.
- Optionally Stops to determine when to close a position complementing the MetaSentimentor. This is a necessity if the MetaSentimentor is not allowed to close or reverse positions.
- Various detailed settings that define how to convert signals into trading actions, what slippage to assume, etc. These specific settings are defined in the so-called Evaluator dialog accompanying a trading approach.

10.3 Defining the Parameter Ranges

As pointed out in Section <u>"TradingSystems – An Overview"</u> the building blocks should reflect a precise trading idea. Moreover, all parameters should be assigned ranges so that the trading idea is maintained.

To adjust the parameter ranges checkmark the Display backtest controls cell in the Tradingsystem Settings section of the DesignerBar. This will display an additional portion of the DesignerBar holding various controls for optimization process:



Click on a parameter of a sentimentor, e.g., the Span parameter of the Kaufmann's Moving Average as shown in the screenshot above, and the parameter as well as its min/max values are displayed and can be adapted.

De	sigr	erBar 💠							ņ
9	× +	🖌 🗙	*	8	📇 🧈	\$+ 🛛	w 💷 🛛		
⊳	Tra	ading							
\triangleright	Br	ackets							
\triangleright	Та	ctic But	tons						
\triangleright	Fil	ters							
- 4	Inc	licators							
	\triangleright	Trading	g			80	, 20, 50, 60	.4	
	\triangleright	Meta S	entime	entor		1, 1	1		
	14	Kaufm	ann's/	AMA		5			
	1	Spa	an			5			
- 4	Tra	adingsy	stem	Settin	gs				
	Tra	ading Ap	proac	h		Fu	ture Tradin	g	
		Equity	Chart						
	~	Display	y back	test c	ontrols				
	Н	MetaSe	entime	entor o	an clos				
	\leq	Go Lon	ig 						
		Go Sho	n ago						
	MC	ire Seun	igs						
E									
a	ʻarar ctua	neter Spa I:	an: min:		max:	Γ	Reset		
5	5		2		25	*	Constant		
Ľ						* L	Constant		
E E	Evaluation Time Range								
fn	om:	02.06 0	0:00	▼ +	to: 17.0	6 19:2	24 - ‡		
v	k?	~ ? ~	\$	n ,		ail	Freeze 🔽		

Note that whenever you enter a value for the parameter value directly in the Indicators section of the DesignerBar then the min/max values of the parameter are automatically updated to make the new setting valid, .e.g, if the Span from the example above is set to 15 then the Max-value is also set to 15:



 Indicators 	
Trading	80, 20, 50, 60.4
Meta Sentimentor	1, 1
 Kaufmann's AMA 	5
Span	30
A Tradingsystem Settings	
Trading Approach	Future Trading
Equity Chart	
 Display backtest controls 	
MetaSentimentor can clos	
✓ Go Long	
Go Short	
More Settings	
Parameter Span: actual: min: max: 30 ‡ 2 ‡ 30	A ★ Constant
Evaluation Time Range from: 02.06 00:00 • ‡ to: 17.06 Image: State St	i 19:24 → ‡ ill Freeze ✓

10.4 Visual Feedback on Parameter Changes

When clicking a parameter of a sentimentor inside the DesignerBar the corresponding sentimentor becomes the *active sentimentor*. This means that the evaluation for this sentimentor is displayed in the InfoBar. Moreover, the signals generated by this sentimentor are displayed in the MasterChart and in the standalone views of the MasterChart.

When changing the current value of the parameter, the active sentimentor is immediately evaluated and the sentimentor as well as its signals are redrawn ("Instant Backtesting").

Clicking inside a sentimentor chart also makes this sentimentor the *active* sentimentor.

10.5 Reset of a Parameter

By clicking on the Reset-button, all three parameter values are reset to their default values.

Paramet	er Span:			
actual:	min:	max:	_	Reset
30	÷ 2	÷ 30	*	Constant

10.6 Fixing a Parameter

For the optimization it is sometimes desirable to fix a parameter at a certain value. This can be done easily with the Constant-button. When pressing this button, the upper and lower bounds of the parameter are set to the current value of the parameter. Thus the parameter can't be varied by the optimization.



10.7 Resetting all Parameters

To reset all parameters of the sentimentors and the actual trading approach to their default values rightclick on any sentimentor and choose Reset All Sentimentors from the context menu.

 Indicators 	h ri / T
Trading	80, 20, 50, 60.4
Meta Sentimentor	1,1 H H
Kaufmann's AMA	م اللعبية العام الم الم الم الم الم الم الم الم الم ا
Span	Visualization
Tradingsystem Settings	Display the sentimentor as subwindow of the MasterChart window
Trading Approach	Aggregation
Equity Chart	Z Edit interpretation
 Display backtest controls 	× Remove Active Sentimentor
MetaSentimentor can clos	🕚 Reset Active Sentimentor
 ✓ Go Long ✓ Go Short 	Save current sentimentor settings as default
More Settings	Sensitivity of the Sentimentor
	Tabu Search for active Sentimentor
Parameter:	Exhaustive search for active Sentimentor
actual: min: max:	★ Fix active Sentimentor
· · · · · ·	Export Active Sentimentor
Evaluation Time Range	C Reset All Sentimentors
from: 02.06 00:00 + 📮 to: 1	📌 Fix All Sentimentors
184 Ank 5.2	× Remove All Sentimentors

10.8 Resetting all Parameters of the active Sentimentor

To reset all parameters of a sentimentor rightclick on it and choose Reset Active Sentimentor context menu.

10.9 Exporting the Active Sentimentor

With the context menu function Export Active Sentimentor the sentiment series of the currently activated sentimentor can be exported into a file. This file can be included in other studies as an "Exported Sentimentor", e.g., for expressing intermarket dependencies, and it can be used in other software.

10.10 The Meta Sentimentor

The Meta Sentimentor usually computes the weighted sum of the sentiment series of all the other sentimentors currently loaded. This weighted sum is used to compute a moving average. The parameters of the Meta Sentimentor define the weights to use for the respective sentimentors as well as the speed of the moving average.

NanoTrader automatically adapts the number of weight parameters whenever an additional sentimentor is added or a sentimentor is removed.

As an alternative to computing the Meta Sentimentor as the weighted sum of the individual sentimentors, it can be defined by logica*l* conditions that are explained in a separate <u>section</u>.



10.11 Long / Short Positions

Checkmark the Go Long / Go Short entries to allow the system to go long and/or short.

al.	Tradingsystem Set	ttings		
	Trading Approach		Future Trading	
	Equity Chart			
	 Display backtes 	t controls		
	MetaSentimente	or can clos		
Г	✓ Go Long			
L	 Go Short 			
	More Settings			

Depending on the traded security you might just want to follow long signals, e.g, when trading stocks.

10.12 Displaying the Equity-window

The development of the equity when trading the generated signals can be visualized in the EquityChart. Just checkmark the corresponding entry:



The equity curve is very important for determining the quality of the generated signal series. The curve should be smooth and ascending without significant drawdowns.

Rightclick on the legend of the Equity chart for more options:





In case the current trading approach is Trendsignals, the Equity-window displays the cumulated evaluation points.

10.13 Using the "Safety Net" for Automatic Trading

NanoTrader allows you to activate a Safety Net for studies that are set to trade in AutoOrder mode. The Safety Net will flatten your position and discard all new signals once a specified condition is met. The Safety Net is implemented as a Filter and hence can be added to a system in the standard way through the Add Sentimentor dialog:

Add Sentimentor	>>	Crossing MA Volume Viewer Exp. Moving Average Time Price Opportunities Key Price Levels Volume Profile LiveStatistics Express\PrevDayClose Bollinger Bands Kaufmann's AMA Meta Sentimentor Express\EMA Trailing Stop Trailing Stop EoP MACD	
Display in MasterChart Display as subwindow in MasterChart windo Insert as: Sentimentor Eiter	Sto <u>p</u>	Tactic Close	

The Safety Net provides the following settings:



al.	Filters		
	Safety Net	1000, 1000, 1000, 0, 0	
	Max Loss	1000	
	Max DrawDown	1000	
	Profit Target	1000	
	Profit Protection Target	0	
	Profit Protection DrawDown	0	

When AutoOrder for the study is activated, NanoTrader monitors the P/L generated by the study. As soon as the P/L exceeds one of the specified numbers the Safety Net gets active, i.e., it closes the current position and discards all new signals generated by the study.

- Max Loss: The maximum tolerated absolute loss.
- **Max DrawDown:** The maximum tolerated drawdown of the P/L. This works like a TrailingStop based on the P/L curve. The drawdown is calculated with each tick (not only at the end of trades).
- **ProfitTarget:** Once the system generated the specified profit it is deactivated to secure the profit.

The **Profit Protection** settings, are interpreted in the following way:

Consider the following example:

a,	Filters		
	4	Safety Net	500, 750, 1000, 800,
		Max Loss	500
		Max DrawDown	750
		Profit Target	1000
		Profit Protection Target	800
		Profit Protection DrawDown	200

If the profit reaches the Profit Protection Target of 800 then the Max DrawDown is tightened from 750 to the Profit Protection DrawDown of 200, starting from the current position profit. This allows to secure an interim profit even if the intended Profit Target is not reached.

If the Protection settings are zero then they are ignored.

The numbers represent cash values in the currency that the P/L for the traded symbol is shown in in the account, i.e., it might be EUR for European futures or USD for American futures.

The Safety Net is only active if it is checkmarked in the DesignerBar. In case you checkmark it *after* the study was set to AutoOrder mode it still measures the P/L since the beginning of the AutoOrdering. This means it could immediately deactivate the study.

If the Safety Net was triggered NanoTrader will show a corresponding message:



Safety Net Triggered	×
Safety Net stopped AutoOrder execution. Reason: Profit Target reached Symbol: Simulation Study: TradeGuard Time: 26.06.09 Fri 13:25	
OK	

The chart background will show the color representing the FLAT state from that moment on:



To let the study start AutoOrdering again just select again the AutoOrder state – this will reset the Safety Net internals:



Note that the Safety Net is only active in AutoOrder mode, not within standard backtesting. For backtesting including optimization you should rely on the Constraints of a study.

10.14 Creating a Report for the Trading Approach

By clicking the Report-button III of the DesignerBar -dialog an HTML report of the actions resulting from the signals of the active sentimentor can be created.



Make sure to activate the MetaSentimentor if you want a report of the complete system.

11 Evaluator-Settings Dialog

The Evaluator dialog can be opened through the 🕮 icon or by clicking the More Settings entry:

11.1 Future Trading Evaluator

11.1.1 Slippage and Transaction Costs

Evaluator Settings	>
Future Trading Trading Const	traints Control Period
Value: 25 Halftrip Costs: 2500 Slippage: 0	per full point Edit Value Ticks
Margin: 16000 Ignore losing trade if opened with of the evaluation period ("warm- Signal execution	nin 1 periods preceding the end up phase").
Sentiment-Enter Signal:	Close same bar 🔹
Sentiment-Exit Signal:	Close same bar 🔹
Execution of stop signal:	Immediately -
	OK Abbrechen

For backtesting a given trading system NanoTrader takes transaction costs and slippage into account. The so-called slippage is a penalty describing the *intended price* and the estimated *execution price*. Assume the system issues a Market buy order when the market currently trades at 4000. There is a good chance that your order will *not* be filled at 4000 but, say, at 4001. In reality you might also receive a better fill, but for backtesting it is save to assume a slippage against you.

As a general note: Whenever there is a choice in the evaluation, NanoTrader will always assume the worst case, i.e., it will never assume the market will

Financial Performance Technology

work for you. This results in a conservative evaluation suiting your needs much better than pretending the market is a good friend of yours. As an example: If the historical data for a, say, 60-minute chart says that both a profit target and a stop are triggered within the same period, then NanoTrader will assume that the stop has been executed, not the profit target.

11.1.2 The "Warm-Up Phase"

Because of the slippage and commission a freshly opened trade will always be a losing trade. Thus, if a trade is opened near the present, it will influence the overall performance negatively. As the optimization tries to maximize the overall performance, it may detect a parameter setting that gets rid of this trade. However, opening signals very near the present are the most important ones. For this reason, NanoTrader allows to define a so-called *warm-up phase* near the end of the evaluation period, i.e., if a trade opened within a number of periods preceding the end of the evaluation period is still a losing trade, then it will not be taken into account for computing the overall performance.



Such a trade is highlighted in the equity curve:

Here, the last trade is still within the warm-up phase shown in blue. The warmup phase is only highlighted in blue if a trade is actually opened in this phase.

11.1.3 Execution of Signals

NanoTrader opens a position always due to a corresponding sentiment, optionally requiring a confirmation in the next period. In contrast, a signal for closing a position can result from a respective sentiment, a triggered *stop loss* or a reached *profit target*.



A signal created for a given period is converted by NanoTrader into a corresponding trading activity. You may choose between certain policies for executing signals:

Sentiment Enter Signals

Signal execution		
Sentiment-Enter Signal:	Close same bar	-
Sentiment-Exit Signal:	Close same bar	
2	Open next bar	N
Execution of stop signal:	Confirmation price next bar	13
	Limit price next bar	

Signal execution	Explanation	
Close same bar	The entry signal is traded at the closing price of the bar that generated the signal (plus slippage and commission).	
Open next bar	The entry signal is traded at the opening price of the bar following the bar that generated the signal (plus slippage and commission).	
Confirmation price next bar	A long entry signal is traded in the bar following the bar that generated the signal, if a confirmation price is reached or exceeded: By default, the trigger price is the highest price of the signal period. In case the opening price of the next bar exceeds the confirmation price, the opening price is used for the trade, otherwise the high of the signal bar is used (always plus slippage and commission).	
	Examples: The opening exceeds the high of the signal bar. Thus, the signal is confirmed and the trade is entered at the opening price.	



For details concerning the Sentiment Enter Signals in conjunction with automatic order execution please refer to the NanoTrader – Charting and Trading manual.

Sentiment Exit Signals



Signal execution	
Sentiment-Enter Signal:	Close same bar *
Sentiment-Exit Signal:	Close same bar 🔹
Execution of stop signal:	Close same bar
	Open next bar

Signal execution	Explanation
Close same bar	The position is closed at the closing price of the bar that generated the signal (plus slippage and commission).
Open next bar	The position is closed at the opening price of the bar following the signal bar (plus slippage and commission).

Stop Signals

A stop signal is either triggered by a stop or by a profit target. In a given bar, potentially both stops could be triggered. As NanoTrader cannot determine the real sequence of ticks in that bar, it always simulates the trades to the disadvantage of the trading result, i.e., a stop signal takes precedence over a profit target signal.



Signal Execution	Explanation
Immediately	In case the current bar reaches the stop or the profit target the position is automatically closed at the trigger price (plus slippage and commission).
	In case the stop or the profit target is reached at the opening of a bar, the position is closed at the opening price (plus slippage and commission).
Close same bar	To determine whether a stop or a profit target has been reached, the close price of the actual bar is taken. In case a signal is triggered, the position is closed at the close price of the actual bar (plus slippage and commission).
Open next bar	To determine whether a stop or a profit target has



been reached, the close price of the actual bar is
taken.
In case a signal is triggered, the position is closed at
the opening price of the next bar (plus slippage and
commission).

11.2 Performance Trading Evaluator

11.2.1 General Settings

Evaluator Settings	×
Performance Money Manage	ement Trading Constraints Control Period
Initial Account Size: 0 Fixed Commission: 0 Variable Commission: 0 Slippage: 0.	Percentage Trading
Ignore losing trade if opened with of the evaluation period ("warm-u Signal execution	nin 1 periods preceding the end up phase").
Sentiment-Enter Signal:	Confirmation price next bar *
Sentiment-Exit Signal:	Close same bar +
Execution of stop signal:	Close same bar *

The trading strategy *Performance* starts with a budget as defined in the Initial Account Size-field. When following a long signal all the cash is invested to buy a maximal number of shares of the analyzed security. The commission to be paid for a trade can be specified through the Fixed commission-field and the Variable commission-field. The latter is defined in percent of the trade volume. Each transaction takes the Slippage into account.

In case the Initial Account Size is set to zero, the wins and losses are calculated in percentages. When applying this "percentage trading" the commissions should also be set to zero. The button Percentage Trading automatically performs these settings. Using Percentage Trading simplifies the comparison of the performance of different titles.

11.2.2 Money Management

Through the rules on the MoneyManagement-page you can define how much of the available cash shall be invested in a trade. (In case of the above mentioned Percentage-Trading, the settings on the MoneyManagement-page are ignored.)

The policy for computing the money to be invested in a trade dramatically influences the overall success or failure of a trading system. For studying Money Management and Risk Management see, e.g., *Van K. Tharp, "Trade your way to financial freedom"*.



Evaluator Settings	×
Performance Money Manageme	nt Trading Constraints Control Period
Invest from the account	25 % per trade
but not more than absolute	4000
and not more than	5 % with a tolerated trade loss of 25 %
and not less than	0
rounded down to a multiple of	50
No trading with negative account	
Calculator	200
available cash: 55	000
current price: 34	
resulting size to be bought:	Calculate
	OK Abbrechen

The settings shown above will be used to calculate the number of shares to be bought as follows:

The current cash is \$ 55000 and we have a Long signal. The price of the security to be bought is \$34.

25% of the cash is \$ 13.750

This amount is higher than the maximal amount of \$4000 we want to invest per trade. Thus, only \$4000 are considered to be invested.

As a second upper bound for the money to be invested, we want to ensure that in case the trade becomes a losing trade not more than 5% of the capital is lost if the losing trade is closed with at most 25% loss.

5% of the capital = 2750. Hence trade volume * 0.25 <= 2750, therefore trade volume <= 11.000.

Using the second risk view point, we could invest up to \$11.000, which is again larger than our limit of \$4000 – so finally we invest \$4000.

With a current price of \$34 and with taking the commissions into account this leads to 116 shares. Rounded to multiples of 50 we end up with buying 100 shares.

In case the field No trading with negative account is checked, a trade is only entered if the minimum required cash defined under and not less than is available. If this field

is unchecked, then any trade is entered using at least the minimal invest, even if the capital becomes negative.

Using the Calculator it is very convenient to compute the shares to be bought for a given capital and security price.

11.3 The Trendsignal -Evaluator

aluator Settings	<u>></u>
Trendsignals	
Unit for Profit/Loss specification: -	
Percent ATR ca	lculated for 20 👘 periods.
Long Signals	- Short Signals
Evaluate Long Signals 🔽	Evaluate Short Signals
Maximal Duration: 8	Maximal Duration: 8
Minimal Profit: 4 00	Minimal Profit: 4 00
Maximum Loss: 1.00	Maximum Loss: 1.00
Correct Signals: 281	Correct Signals: 344
Signal Filter	
Value only one signal pe	er 5 🗧 periods.
Remaining Long Signals	× 112
Remaining Short Signals	× 114
Points per	
correct Long signal: 1	correct Short signal: 1
wrong Long signal: -1	wrong Short signal: -1
Best possible rating	: 226 C <u>a</u> lculate
	<u>R</u> eset
	OK Abbrechen

The page of the *Trendsignals*-evaluator is structured in four parts.

Definition of the unit for profits and losses:

For specifying the unit of profits and losses, NanoTrader allows to choose between percentages, multiples of the Average True Range ("ATR"), and absolute prices. When choosing "ATR", the specified number of periods is used for calculating the respective ATR per bar. In this case, the investment goals are automatically adapted to the volatility of the defined time range. The *Average True Range* is a moving average on the *True Range* which is defined as the largest difference of

- today's high minus today's low
- today's high minus yesterdays close
- yesterdays close minus toady's low



Definition of the Investment Goal:

The groupboxes Long-Signals and Short-Signals are used to define investment goals for long and short positions in the underlying.

- Long Signals	- Short Signals
Evaluate Long Signals 🔽	Evaluate Short Signals 🔽
Maximal Duration: 10 📩	Maximal Duration: 10
Minimal Profit: 6 🔺 %	Minimal Profit: 10 🔹 炎
Maximum Loss: 2 🔹 🕺	Maximum Loss: 2 🔹 %
Correct Signals: 167	Correct Signals: 91

Long and short signals can be generated independently of each other. By activating or deactivating the Evaluate Long-Signals and Evaluate Short-Signals checkboxes, you define the types of signals to be generated. The best results are achieved if just one type of signal has to be generated.

To decide whether a generated trendsignal is *correct*, the following approach is used: A long trendsignal is considered correct if within Maximal Duration days the quote of the underlying raises at least Minimal Profit percent without hitting the stop loss limit of Maximum Loss percent. Otherwise, the trendsignal is considered *wrong*.

The correctness of short trendsignals is determined analogously.

The number of points in time which would have led to a correct signal is displayed in the fields Correct Signals. Moreover, these correct signals are visualized in the MasterChart. Whenever the investment goal is changed, the number of correct signals and the visualization are updated. This allows to verify conveniently how often the current underlying did meet the actual investment goal.

Signal Filter

It happens quite often that within a small time period many correct signals could be created. If the optimization finds a parameter setting that generates signals for such a particular time range, the rating becomes very good and possibly optimal as in the following example:



Unfortunately, this optimal parameter setting does not generate any signal outside this small time range. This happens because a parameter setting which would produce signals outside this time range could create less signals in the "accumulation" period, and thus produces an overall worse rating.



The Signal Filter settings let you overcome this effect:



In the example it is specified that within fifteen days at most one correct (or wrong) signal will be used for the evaluation. This results in an optimal solution which is of much more interest:



Evaluation Points

The evaluation assigns points to each generated signal.

- Points per-	1
correct Long signal: 1 correct Short signal: 1	
wrong Long signal: -2 wrong Short signal: -2	
Best possible rating: 25 Calculate	

In this example each correct signal generated by a sentimentor will be valued with one point, whereas a wrong signal is valued with –2 points, i.e., a wrong signal neutralizes two correct signals.

NanoTrader displays the best possible rating that can be achieved if all correct signals are found and no wrong signal is generated. After changing the valuation points, the best possible rating can be recalculated by pressing then Calculate-button.

Resetting the Parameters

By clicking the Reset-button all evaluation parameters are reset to their initial values. This allows for easy experimentation with investment goals, valuation points, and the filter.

12 Evaluation Time Periods

12.1 General Notes

NanoTrader allows to partition the loaded price data into three time ranges that are specifically interesting when developing automated trading strategies:

• Optimization Time Range

The price data inside this range is the primary basis for an optimization. NanoTrader maximizes the chosen optimization goal, e.g. the net profit, on this data. Restrictions can be used in the optimization process to rule out systems that show unwanted characteristics like a too large drawdown.

• Control Time Range

The control time range is used to check the results of a parameter setting found in the optimization range on "unknown" data. Restrictions can also be defined for the control period that have to be obeyed in the optimization process. As an example, the optimization time range could be optimized for a maximal net profit and a restriction could express that the profit factor in the control time range should be at least 1.5.

Tail Time Range (Out of Sample Data)
 In contrast to the control time range that influences the optimization process indirectly through restrictions, the tail time range remains completely unknown for the optimization. Hence, the tail time range allows to see immediately in terms of graphics and statistics the behavior of a system on completely unknown data.

The partitioning of the price data into the three distinct time ranges helps greatly in the development of stable, robust trading systems.



12.2 Visualization of the Time Ranges

The Equity chart visualizes the different time ranges as shown above.

12.3 Specifying the Evaluation Time Range

By specifying the evaluation time range the optimization and control time range are automatically set, as the control time range is defined as a certain percentage of the evaluation time range. In the screenshot given above, the control period is defined as 25% of the evaluation time range. Through this coupling, the different ranges can be set and changed very conveniently.

To specify the time range for evaluation time range, use the controls of the Evaluation Time Range-groupbox:



The evaluation time range can be specified explicitly using the From/To-date pickers.

The Get-button Allows to make the actual zoom range the evaluation time range.

After changing the time range explicitly, you can use the Zoom to Evaluation.button ... to zoom the windows to back to the evaluation time range.

Using the All-button ^{Source}, the evaluation time range is set to cover the complete loaded data.

When saving a study only the From-date is saved. The To-date is automatically set to the latest available date when loading the corresponding quote file. This enables an easy daily, say, evaluation of stocks using batch files without the necessity to adapt the time range manually.

12.3.1 Specifying the Control Time Range

The control time range is defined as a percentage of the evaluation time range. The percentage is defined on the page Trading:



Evaluator Setting	s		×
Future Trading	Trading Constraints	Control Period	
Optimization Go	oal:		
Total Performa	ince	-	
0 <=	total number of trad	es <= 10000	
0 % <=	largest winning trade * Total Net Profit	100% <= 100000	%
0 <= a	avg. number of bars in w	vinners <= 100000	
	absolute maximal draw	down <= 1000000	
	relative maximal draw	down <= 100000	%
	max. consecutive	losers <= 10000	
0 % <=	number of winning tra- total number of trad	<u>des * 100%</u> les	
0 <=	profit factor		
-10000(<=	Net profit		
	max. Parameter S	catter: <= 10000	%
	avg. Parameter S	catter: <= 10000	%
Con	trol Periods 25 % of	ps of Evaluation Period	
		No Constraints	
		ОК	Abbrechen

In case the control time range is not required specify 0.

12.3.2 Tail Time Range

The tail time range is automatically set by NanoTrader to the time range following the evaluation time range. The tail time range may be empty.

12.4 Switching to the Tail Time Range

NanoTrader allows to switch between the evaluation time period and the tail time period by clicking the **Lail**-button. This button is clickable if the current evaluation time period does not consume the end of the available price data.

 Evaluation 	ation Time Range —		
from:	16.06 09:50 👻 🌲	to:	17.06 02:59 🔹 🗘
1	. ∛ ≹		Tail Freeze 🗸

When clicking the \Box Tail -button, NanoTrader switches the evaluation time range internally to the *tail*, i.e. from 3/4/02 to the end of the available data, and displays the trading results in terms of trading signals and performance report. To switch back to the original evaluation time range, just click again on the \Box Tail -button.

13 Optimization Goals and Constraints for Trading Strategies

13.1 General Notes

The goal of NanoTrader is to help in creating trading strategies that are as *good* as possible. The notion *good*, however, must always be related to the concrete scenario the strategy is designed for as well as to the general conditions defined by the trader. On first sight a strategy is "good" if it creates much profit, but when looking deeper into the strategy it may well be that the high profit has the price of a high and unacceptable drawdown. It may also be the case that the profit is high, the drawdown is low, but the strategy created a series of 10 losing trades that are very hard to take from a psychological point of view. Therefore, a strategy has to be both *good* and *acceptable*.

NanoTrader measures the *goodness* of a strategy by a so-called *optimization goal*, sometimes also called the *objective function*.

To express what strategies are acceptable so-called *constraints* are used, .e.g, "accept only strategies that result in at least 10 and at most 30 trades in the given evaluation period".

The task of the optimization is to find a parameter setting with a maximal objective value that respects all constraints. Parameter settings violate a constraint are not considered even if they show a higher objective value.

The optimization goal and the constraints are defined on the Trading Constraints page of an Evaluator:



Evaluator Settings ×
Future Trading Trading Constraints Control Period
Optimization Goal:
Total Performance 🔹
0 <= total number of trades <= 10000
$0 \qquad \% <= \frac{\text{largest winning trade * 100\%}}{\text{Total Net Profit}} <= \frac{100000}{\%}$
0 <= avg. number of bars in winners <= 100000
absolute maximal drawdown <= 1000000
relative maximal drawdown <= 100000 %
max. consecutive losers <= 10000
0 $\% \ll 100\%$
total number of trades
0 <= profit factor
-10000(<= Net profit
max. Parameter Scatter: <= 10000 %
avg. Parameter Scatter: <= 10000 %
using 10 Steps
Control Periods 0 % of Evaluation Period
No Constraints
OK Abbrechen

13.2 Optimization Goals

NanoTrader provides the following choice of optimization goals:

- total performance
- average winning trade
- percent profitable trades
- Expectancy, defined as (PW x AW- PL x AL) / AL with:
 PW = probability of winning AW = average winning trade
 PL = probability of losing AL = average losing trade
- Expectancy Score, , defined as [(PW x AW- PL x AL) / AL] * Opportunity with: PW = probability of winning AW = average winning trade



PL = probability of losing AL = average losing trade Opportunity = # Trades / # Periods in the evaluation period

- max consec winners max consec losers
- Profit Factor
- total performance / max drawdown
- Happiness-Factor , by Stefan Fröhlich (English Happy = German Fröhlich), defined as: [Performance * PW * (ProfitFactor + AW / AL)] / [max. drawdown + max. losing trade + max winning trade]
- Average Trade

Because of their definition many optimization goals require additional constraints, e.g., the maximal possible percentage of profitable trades is 100% - this could be reached by a strategy creating exactly one winning trade. To exclude such "pathological" cases, a minimum required number of trades and/or a minimum performance should be defined as constraints.

13.3 Defining Constraints

A restriction defines a lower and upper bound for a classification number such as profit factor or number of trades. In case the classification number lies outside this range, the constraint is *violated*.

NanoTrader requires a parameter setting to comply with all constraints.

The default constraints are chosen such that they are not sharp – i.e. every parameter setting of the sentimentors will be feasible.

By clicking the No Constraints button the restrictions are reset to their defaults.

A wise proceeding in working with constraints is to start with the default settings. After an optimization you check whether your personal preferences are met by the computed parameter setting. If not, you sharpen the respective constraints.

Of course it may happen that there exists no sentimentor parameter setting at all that produces a series of signals complying with the constraints.

A parameter setting leading to a violation of the constraints is visualized by a red button in the DesignerBar:



13.4 Optimizing with Embedded Sensitivity Analysis

The constraints max Parameter Scatter and avg Parameter Scatter allow to include results from the sensitivity analysis as described in a dedicated <u>section</u>.



Evaluator Setting	s	×
Future Trading	Trading Constraints Control Period	
Optimization Go	pal:	
Total Performa	nce *	
0 <=	total number of trades <= 10000	
0 % <=	largest winning trade * 100%<=100000Total Net Profit%	
0 <= a	avg. number of bars in winners <= 100000	
	absolute maximal drawdown <= 1000000	
	relative maximal drawdown <= 100000 %	
	max. consecutive losers <= 10000	
0 % <=	number of winning trades * 100% total number of trades	
0 <=	profit factor	
-10000(<=	Net profit	
	max. Parameter Scatter: <= 10000 %	
	avg. Parameter Scatter: <= 10000 %	
	using 10 Steps	
Con	trol Periods 0 % of Evaluation Period	
	No Constraints	
	OK Ab	brechen

In case one of these constraints receives a sharp upper bound NanoTrader will conduct the sensitivity analysis for every considered parameter change.

Caution: Usually this leads to a significant increase of required computation time.

The rationale behind this is the assumption that the higher the sensitivity of the parameters, the less likely it is that this setting will work on unknown data. By constraining the parameter sensitivity, NanoTrader automatically attempts to compute more robust parameter settings.

13.5 Optimization including the Control Period

A detailed description of the control time period can be found in section <u>Evaluation Time Periods</u>.

On the page Control Period restrictions can be defined that are verified for the control time range. The optimization rejects all parameter settings that do not obey these restrictions.

14 Optimization of a Tradingsystem

14.1 General Notes on Optimization Techniques

As pointed out in section <u>Optimization and Robustness</u> each trading system has a number of parameters that can vary in the range of their respective minimal and maximal values. Using straightforward combinatorial calculations you will see that even small systems will have literally billions of potential parameter settings. The simple system described in the introduction consisting of a Moving Average, a MACD, and a Trailing Stop already created 312.500.000 possible parameter combinations. Assume just one more parameter ranging between 10 and 25 and the combinations are increased to 5.000.000.000.

Obviously an exhaustive search, i.e., trying every parameter setting, is not possible. This implies also that there is no general way to find an optimal parameter setting. Reality is that we need to be content with a so-called "near optimal" parameter setting. Exhaustive search though is supported by NanoTrader for individual parameters or sentimentors.

In order to find a very good or even optimal setting from these enormous number of possibilities NanoTrader uses a specific optimization technology derived from the so-called *Tabu Search*. Tabu Search is an extremely efficient heuristic algorithm allowing to find a very good parameter setting in a very short time. A specific advantage of Tabu Search is the possibility to use this approach for any kind of sentimentor or MetaSentimentor, because it adapts dynamically to the given environment.

If you compare the NanoTrader optimization technique with other approaches like Genetic Algorithms, you will see that NanoTrader by far outperforms these techniques, i.e., you will get a better solution in a shorter time. Only because of this superior optimization technology the additional components such as Walkforward Optimization or Sensitivity Analysis can be made an integral part of the optimizer.

NanoTrader allows you to optimize an individual parameter, an individual sentimentor, or the complete system at once. There is *no* limitation on the number of parameters a system can have.

14.2 Optimizing a Single Parameter

An individual parameter will usually be optimized by exhaustive search, i.e., the MetaSentimentor, and thus all building blocks of the study, is used for computing the signals. However, only the active parameter will be varied through all its feasible settings.

To optimize a single parameter rightclick on its name and choose Exhaustive search for active Parameter from the context menu:





14.3 Optimizing a Single Sentimentor

To start the optimization for an individual sentimentor click on that sentimentor in the DesignerBar and then click the Optimize icon set of the DesignerBar:

Germany 30 CFD [10 Min.]	10_Min	×		
DesignerBar		000000000000000000000000000 P	🖽 [10 Min.] Germar	
🔊 - 🖌 🗙 🔅 🔀	A -	\$+ ₩ 🏢 🖬	Buy Sell	
 Trading Brackets 	4	✓ Optimize		
 Diackets Tactic Buttons 		Optimize the ad Parameter with	tive indicator or active TabuSearch. If the	
 Filters 		MetaSentimentor is activated the		
✓ Flat	18:00 -	complete study	will be optimized.	

Alternatively rightclick the sentimentor and choose Tabu Search for active Sentimentor.

In case the sentimentor to be optimized is a Stop or a Filter then the MetaSentimentor, and thus all building blocks of the study, is used for computing the signals. However, only the parameters of the sentimentor to be optimized will be varied.

In case the sentimentor to be optimized is a "normal" sentimentor, i.e., *not* a Stop, a Filter, and *not* the MetaSentimentor, then only the sentiments of this sentimentor are the basis for the signal generation. The Stops and Filters are applied on the sentiments of this sentimentor.

If NanoTrader encounters two parameter combinations which result in the same optimal evaluation, it will keep the combination that produces a signal nearer the present so you might decide to follow that "fresh" final signal.

When choosing an Exhaustive search for active Sentimentor from the context menu NanoTrader will evaluate each possible parameter setting of the active sentimentor. This approach should only be used if the number of feasible parameter combination is rather small (< 1 million).

14.4 Optimizing the Complete Tradingsystem

All parameters of a given tradingsystem will be optimized if the MetaSentimentor is selected for optimization:



It is not necessary to find a specifically good starting parameter setting manually – usually Tabu Search will find a much better setting in less time. However, the result of an optimization should be saved and can then serve as a starting point for subsequent optimization runs.

15 The Optimize Dialog

As described above the Optimize-dialog can be started by first activating a sentimentor and then clicking the 🛩 icon in the toolbar of the DesignerBar:

Alternatively, right-click on the sentimentor and select TabuSearch for active sentimentor from the context menu:

The Optimize dialog looks as follows:



Progress		Current Best Rating				
Tries	: 938	ltem	Value	Control	Tail	
15 %	6	Total net profit:	120.10	10.90	157.20	
Improves	· 6	Total # of trades:	3	1	3	
Last impo	ovement at try:	Winning trades:	3	1	2	
cust impro	C07	Losing trades:	0	0	1	
	037	Percent profitable:	100.00%	100.0	66 67%	•
Been :				hanned to be a set of the set of		
Start	Stop C	lose Confirm Close	Evaluator	·/•		
Start Ranking	Stop C Optimization Period	lose Confirm Close	Evaluator Tail Period	• ^ •		
Start Ranking	Stop C Optimization Period 120.10	lose Confirm Close Control Period 10.90	Evaluator Tail Period 157.20			ĺ
Start Ranking 1 2	Stop C Optimization Period 120.10 118.20	lose Confirm Close Control Period 10.90 10.90	Evaluator Tail Period 157.20 151.20			
Start Ranking 1 2 3	Stop C Optimization Period 120.10 118.20 96.20	lose Confirm Close 10.90 10.90 10.90 10.90	Evaluator Tail Period 157.20 151.20 27.10			

The title of the Optimize dialog displays the component which is being optimized.

By clicking the Start-button the optimization is started. To interrupt the optimization press the Stop-button. While optimizing the Optimize-dialog displays continuously the progress.

While an optimization is running, the title bar of the Optimize-dialog displays the starting time. After finishing the optimization, it displays the consumed runtime.

15.1 Defining the Stop Criterion

The criterion for stopping the optimization can be defined in the Stop aftergroupbox:

Optimization of Meta Ser	ntimentor Runtime: 00:(00:09			×
Progress	Current Best Rating -				
Tries: 938	ltem	Value	Control	Tail	
15 %	Total net profit:	120.10	10.90	157.20	
Improves: 6	Total # of trades:	3	1	3	
Last improvement at try:	Winning trades:	3	1	2	
co7	Losing trades:	0	0	1	
037	Percent profitable:	100.00%	100.0	66.67%	•
Animation	Stop after				
Show All Tries	max. Tries: 1	0000			
Show Parameters					
Beep at improvement	max. Time: 0	0:01:00 💂	⊻		

The stop criterion is a combination of number of tries to be performed and the runtime. Use the respective checkboxes to activate the criterions. If both criterions are deactivated the optimization will continue until the Stop-button is pressed, or, in case of exhaustive search, all parameter combinations have been tested.

For exhaustive search, max. Tries has a special meaning: Suppose the max. Triescheckbox is checked and max. Tries is set to 1000. If there exist, say, 10000 possible parameter combinations for the sentimentor to be optimized then every



10th combination will be checked. This allows one to check *very different* parameter combinations and not just the first 1000 of the complete enumeration.

15.2 Animation

The animation settings are defined in the Animation-Groupbox:

Animation
Show All Tries
Show Parameters
Beep at improvement

When checking the Show All Tries-checkbox, the sentimentor being optimized is continuously redrawn in its window

Moreover the Show Parameters-checkbox allows to display the current computed parameter setting in the Designer-table.

Activate Beep at improvement to let NanoTrader issue a beep whenever it encounters an improved parameter setting.

The animation settings can be changed even while the optimization is running.

Whenever a new best parameter combination is found it is displayed independently of the animation settings.

The settings for the animation and stopping criterion are stored persistently.

15.3 Progress control

The Progress-groupbox informs about the current state of the optimization.

Progress
Tries: 938
15 %
Improves: 6
Last improvement at try:
697

The Tries are a kind of "pass counter" that is increased continuously. The bar and percentage display show the time/effort already consumed until the stopping criterion will be met. The field Improves displays the total number of encountered improved parameter settings.

The "Try"-number where NanoTrader found the last improvement is displayed in the Last improvement at try-field.

15.4 Optimization History

The lower part of the Optimize dialog shows a history of all found improved solutions. By clicking on a row, the corresponding setting becomes active and the charts, signals, statistics, and DesignerBar are updated accordingly.
Hint: If you make the complete loaded price data visible before you start the optimization, the course of the optimization can followed visually.

Rightclicking into a row opens a context menu that allows to save the corresponding setting as a study:

Progress		Current Best Rating				
Tries	: 938	Item	Value	Control	Tail	
15 %	6	Total net profit:	96.20	10.90	27.10	
Improves	. 6	Total # of trades:	2	1	1	
Last impr	ovement at try:	Winning trades:	2	1	1	
Lust impr	607	Losing trades:	0	0	0	
	037	Percent profitable:	100.00%	100.0	100.0	
⊡ Beep a	at improvement Stop	Close Confirm	00:01:00 ‡			
D 11	Optimization Period	d Control Period	Tail Period			
Ranking			157.00			
Ranking	120.10	10.90	157.20			
Ranking 1 2	120.10 118.20	10.90 10.90	157.20			
Hanking 1 2 3	120.10 118.20 96.20	10.90 10.90 10.90	157.20 151.20 27.10			
Ranking 1 2 3 4	120.10 118.20 96.20 78.70 Save	10.90 10.90 10.90	157.20 151.20 27.10 31.10			
Ranking 1 2 3 4 5	120.10 118.20 96.20 78.70 Save 72.20	10.90 10.90 10.90 1 1	157.20 151.20 27.10 31.10 31.10			
Ranking 1 2 3 4 5 6	120.10 118.20 96.20 78.70 Save 72.20 50.30	10.90 10.90 10.90 517.40 10.90	157.20 151.20 27.10 31.10 31.10 28.60			

16 Sensitivity Analysis

16.1 General Notes

For the evaluation of a given parameter setting (i.e., all the parameters visible in the DesignerBar) it is quite interesting to get information about the *sensitivity* of these parameters. In other words, "How much does the objective function, e.g., the net performance, vary if a parameter is slightly changed?"

The rationale behind this question is the assumption that the higher the sensitivity of the parameters, the less likely it is that this setting will work on unknown data.

NanoTrader can compute and visualize the sensitivity automatically by using the following algorithm:

Starting from the initial parameter setting each parameter is changed subsequently in a range from

-10 * step size + initial parameter value up to +10 * step size + initial parameter value

and the new value of the objective function, e.g., the net performance, is computed. A *step size* equals the change in the parameter when using the up/down arrows in the Parameter group box for the given parameter. Instead of the default 10 any other number of steps can be chosen.

In case a parameter change leads to a value lying outside the feasible range or if the resulting setting violates a trading restriction then this parameter change is not conducted.





By dragging with the mouse the perspective of the 3D-chart can be changed. When pointing with the mouse at a line a popup window will be shown:



Using the selection box at the bottom of the dialog, you may select a different charting mode, e.g., 2D lines:

Financial Performance Technology



The X-axis shows the change in step sizes of the parameters. The 0 location corresponds to the initial parameter setting. The Y-axis shows the value of the chosen objective function, e.g., the net performance.

The footer of the graphic shows some statistics based on the average percentage change of the objective function with respect to the initial setting – this is called the *parameter scatter*.

- the "Average Scatter" is the average over all parameter scatters
- the "Max. Scatter" is the maximal measured parameter scatter
- the "StdDev" is the statistical standard deviation over all parameter changes.

NanoTrader allows to use the results of a sensitivity analysis within the optimization by using corresponding <u>restrictions</u>.

16.2 Sensitivity Analysis of a complete Study

Rightclicking on the MetaSentimentor in the DesignerBar opens the context menu:





Choose System Sensitivity to check the sensitivity of all parameters that

- are not fixed
- are not set to 0 (applying the assumption that a parameter set to zero means "disabled", e.g., TrailingStop)
- have a domain unequal {0, 1} (applying the assumption that such a parameter is a on/off switch, e.g., a certain pattern in the CandleStick sentimentor)

16.3 Sensitivity Analysis of a Sentimentor

Equals the analysis of the complete system, but only the parameters of the selected sentimentor are examined.

16.4 Sensitivity Analysis of a Parameter

Only the active parameter is examined using its complete value range.

16.5 Trade Histogram

Using toolbar button 🔤 of the DesignerBar a dialog showing all trades as a histogram can be opened:





This is view is a very interesting supplement to the equity chart as it shows clearly the relations between winning and losing trades as well as series of winning and losing trades.

Whenever the underlying study is changed of new ticks arrive, the histogram is updated automatically.

The selection box allows to switch to a number of different histogram modes:





Note: The most histogram variants require the equity chart to be activated in the DesignerBar:

DesignerBar	ņ		
🔊 - 🖌 🗙 🌞 🔀 📇 🗸	• \$+ 👐 🏢 🖬		
▷ Trading			
Brackets			
Tactic Buttons			
 Filters 			
Flat			
 Indicators 			
Trading	76, 24, 40, 60		
Meta Sentimentor	1, 1		
 Crossing MA 	12, 52		
Fast MA-Span	12		
Slow MA-Span	52		
 Tradingsystem Settings 			
Trading Approach	Future Trading		
Equity Chart			
Display backtest controls			
MetaSentimentor can cl			
📕 🖌 Go Long			
Go Short			
More Settings			

17 Scripts for Automated Study Optimization

Scripts enable the automated optimization of studies for a number of securities. A typical application of scripts is to let NanoTrader work on the studies in the night. The most promising securities are presented in the morning and can be used as a basis for the trading day.

17.1 Running a Script

The WorkspaceBar displays the scripts stored in the ScriptsDir below the entry Screeners (Scripts).



When doubleclicking a script it will be started.



After starting a script the ScriptViewer comes up. The ScriptViewer is used to monitor the activities while processing the script. A script run can be aborted by clicking the Stop-button of the.

Processing Script 'JWSignalMagic'.		×
Press Start to begin.	Ţ	*
Working on underlying 'Beiersdorf CFD' '1002 Optimization facts: tries= 50 improves= 8 la	05' (17153). stImprove= 39 (0 seconds) totalTime = 0	
Working on underlying 'HeidelbergCement AG Optimization facts: tries= 50 improves= 11 I	6 CFD' '100217' (2/153). astImprove= 37 (0 seconds) totalTime = 0	
Working on underlying 'Merck KGAA CFD' '10 Optimization facts: tries= 50 improves= 10 I	00225' (3/153). astImprove= 50 (0 seconds) totalTime = (
Working on underlying 'Rheinmetall VZ CFD' Optimization facts: tries= 50 improves= 6 la	'100229' (4/153). stImprove= 49 (0 seconds) totalTime = 0	
Working on underlying 'Fresenius Medical Ca Optimization facts: tries= 50 improves= 4 la	re CFD' '130479' (5/153). stImprove= 27 (1 seconds) totalTime = 1	_
4	•	
S	itop	
		:

For each optimized security the number of performed tries, the "try" with the last improvement and the consumed runtime are reported. This information can be very helpful to find a good setting for the stopping criterion defined in the script to be used for the next runs.

17.2 Editing a Script

The ScriptEditor can be started by rightclicking on the script to be edited and choosing Edit in the context menu.





The ScriptEditor looks as follows:



	ic			
Securities to be proc	essed by screer	ier		
Security Name	Symbol	Data Source	Location	-
1&1 Drillisch AG	400767444	CFD-Forex	CFD-Forex	
401129542	401129542	CFD-Forex	CFD-Forex	
Aareal Bank AG	400767005	CFD-Forex	CFD-Forex	
Adidas AG CFD	99594	CFD-Forex	CFD-Forex	
Adva Optical Net	400767088	CFD-Forex	CFD-Forex	
Aixtron SE CFD	400767090	CFD-Forex	CFD-Forex	
Allianz SE CFD	99596	CFD-Forex	CFD-Forex	
Alstria Office REI	401063735	CFD-Forex	CFD-Forex	
Aroundtown SA	401572199	CFD-Forex	CFD-Forex	
Aurubis AG CFD	400767007	CFD-Forex	CFD-Forex	
BASF SE CFD	99598	CFD-Forex	CFD-Forex	
Bayer CFD	99602	CFD-Forex	CFD-Forex	
itop after				
max. Tries per Stor	sk: 50	Approx	ximate Completior	n time
and the per bloc		10.00	2021.07.12	
max.Time per Stoo	sk: 12:58:55	- 12.09	.2021 07:13	× +
max.Time per Stoc	ck: 12:58:55	- [] [12.09	.2021 07:13	* *
max.Time per Stoc Study Settings Template Study: \$	ck: 12:58:55 (DefaultDysDir) ⁽	JWSignalMagic.o	dys Brows	e
max. Time per Stoo Study Settings Template Study: \$ Study-Label: J	ck: 12:58:55 (DefaultDysDir)\ WSignalMagic	JWSignalMagic.o	dys Brows	se
max. Time per Stor Study Settings Template Study: \$ Study-Label: J ¹ he screener will load ot exist for the secur Update existing str	ck: [12:58:55 (DefaultDysDir) WSignalMagic I the study with ity the defined tu udies with templ	JWSignalMagic.o ulwSignalMagic.o the defined label. emplate study is u ate study	dys Brows In case the study ised.	se y doe
max. Time per Stor Study Settings Template Study: \$ Study-Label: J he screener will load ot exist for the secur Update existing study olumn layout for resu	ck: [12:58:55 (DefaultDysDir)\ WSignalMagic I the study with ity the defined to Jdies with templ Jit table:	JWSignalMagic.o the defined label. emplate study is u ate study	dys Brows In case the study ised.	se y doe

17.2.1 Selecting Securities

The securities to be processed by the script can be selected by clicking the Select Securities-button. This will open the following dialog:



Select Symbols	×
	ה
	1
🔺 🗌 盲 CFD-Forex	•
Bonds	
🖻 🗌 Commodities	
Equities	
🖻 🗌 Equities Belgium	
Equities Canada	
🖻 🗌 Equities Denmark	
Equities Europe	
Equities Finland	
Equities France	
Equities Germany	
🔲 Equities Hong Kong	
Equities Ireland	
Equities Japan	
Equities Netherlands	
Equities Norway	
Equities Portugal	
Equities Singapore	
Equities Spain	
Equities Sweden	
Equities Switzerland	
Equities UK	٣
OK Cancel	

The displayed securities vary with the activated data sources.

Securities to be processed by the script are checkmarked. You may change the selection by checking/unchecking the respective securities. It is also possible to mix securities from different data sources, e.g. from MetaStock and quote files.

If a folder is checkmarked then the securities inside the folder are automatically processed by the script – no matter if the securities of the folder carry a checkmark or not.

After quitting the Select Securities-dialog the securities to be processed are displayed in the ScriptEditor:

Aareal Bank AG	400767005	CFD-Forex	CFD-Forex	
Adidas AG CFD	99594	CFD-Forex	CFD-Forex	
Adva Optical Net	400767088	CFD-Forex	CFD-Forex	
Aixtron SE CFD	400767090	CFD-Forex	CFD-Forex	
Allianz SE CFD	99596	CFD-Forex	CFD-Forex	
Alstria Office REI	401063735	CFD-Forex	CFD-Forex	
Aroundtown SA	401572199	CFD-Forex	CFD-Forex	
Aurubis AG CFD	400767007	CFD-Forex	CFD-Forex	
BASF SE CFD	99598	CFD-Forex	CFD-Forex	
3ayer CFD	99602	CFD-Forex	CFD-Forex	-
C			Þ	
			P	+

17.2.2 Defining the Runtime per Security

The optimization spends a certain time per security. This time can be specified through the number of tries or in an explicit duration. Thus, the same mechanism as already described for the Optimize-dialog is used.

Stop after	
max.Tries per Stock: 50 📐 🗌	Approximate Completion time:
max.Time per Stock: 00:05:00 🗘 🔽	22.06.2021 01:47 🔹 🗘

If both stopping criterions are activated, the optimization stops as soon as the first criterion is met.

If a fixed time per security is defined, then automatically the expected completion time is displayed below the Approximate Completion Time-label. This assumes that the script is started immediately.

Very often one likes to run a script immediately and wants the script to have finished at a certain time. This can be achieved easily by editing the completion time. Editing the completion time will automatically adapt the max Time per Stock setting as require:

Stop after		
max.Tries per Stock:	50	Approximate Completion time:
max.Time per Stock:	00:02:07 ‡	21.06.2021 18: <mark>30</mark> - ‡

By leaving the ScriptEditor with the Save & Run-button the script is saved and executed in one step.

17.2.3 Defining the Study Label and Standard Study

For each processed security, a script loads a corresponding study using the label defined in the script (Study-Label).



In the example given above, for each security to be processed, the script will load the study with label "trading". In case such a study does not exist for the security, the template study trading.dys from the subdirectory DefaultDys of the installation directory is loaded.

If the box "Update existing studies with template study" was checked the script will use the defined template study for each security. Even if the study with the defined label is already existing.

The loaded study will be optimized until the stopping criterion is met. Finally, the optimized study will be saved with reference to the security.

Whenever possible, the name of a script, the Default Dys-File and the Dys Label should be the same to make the relations as clear as possible.

17.3 Creating a "Template Study"

To create a Template Study proceed as follows:

- 1. Load an arbitrary chart
- 2. Select the Trading Approach.
- 3. Adapt the Evaluator to suit your needs.
- 4. Select the sentimentors to be used in this study.
- 5. Make sure the upper-and lower bounds of the parameters of the sentimentors and trading approach have meaningful values.
- 6. Choose the start date for the evaluation period.
- 7. Finally click Save As Template Study and store the study using a meaningful name.



17.4 Exporting MetaSentimentors

In the process of running a script, the MetaSentimentors of the processed studies can be exported by activating the check box Sector MetaSentimentors for inclusion in other studies

The filenames used for storing the MetaSentimenors are of type:

security name#study label.sent

The files are stored in the directory "Exported Sentimentors" below the installation directory.



17.5 Creating a New Script

To create a new script start rightclick on Scripts in the WorkspaceBar and select New from the context menu:



The ScriptEditor is opened with an empty script. Now enter a name for the new script:

Screener Editor				×
Name: Securities: e pro	cessed by scre	eener		
Security Name	Symbol	Data Source	Location	
	•			

Now complete the entries in the script as described above and save the script using the Save or Save & Run button.



Normally you want to build on an already existing and optimized study. If this is not the case, then check the following box:

Study Settings		
Template Study:	\$(DefaultDysDir)\Trading.dys	Browse
Study-Label:	Trading	
The screener will lo not exist for the sec Update existing	ad the study with the defined label. In case curity the defined template study is used. studies with template study	the study does

17.6 Creating a New Script from an Existing Script

If you just want to adapt an existing script slightly and store it under a new name proceed as follows:

- 1. Load the existing script into the ScriptEditor.
- 2. Overwrite the name of the script with the name of the new script
- 3. Adapt the script settings.
- 4. Save the script.

17.7 Deleting a Script

Rightclick on the script to be deleted and choose Delete from the context menu.

17.8 Starting Scripts from the Command line

NanoTrader allows to start scripts from the command line. This allows to automate an update of the quote files and a subsequent optimization.

The syntax for running scripts from the command line, e.g., MS-DOS Command Prompt, is as follows:

NanoTrader.exe -script <script 1.dsb> ... <script n.dsb>

To create status reports use the following command:

NanoTrader.exe -status <script 1.dsb> ... <script n.dsb>

The script files must either be specified using the complete pathname or relative to the execution directory.

Example:

NanoTrader.exe -script "C:\Programs\Fipertec\NanoTrader\Scripts\trading.dsb"

17.9 ScreenerBar

At the end of a script run NanoTrader presents a results report in the so-called ScreenerBar.

Screeners										- ×
Name	Rating	+/-	Last Signal	LS Type	Last Confirmation	LS Quote	Act Quote	Act Trade	Act Sentiment	•
Kloeckner & Co CFD	4.12%	64/72	21.06. 13:12	close short (Stop 11.1130 (Trailing Stop EoP))	21.06. 13:20	11.1100	11.1100	n/a	35	
Ceconomy AG CFD	2.45%	26/46	21.06. 13:20	Short (Stop 4.2030 (Trailing Stop EoP))	21.06. 13:20	4.2000	4.2000	0.00	0	
Saf Holland CFD	2.44%	15/10	21.06. 13:12	close short (Profit Target 12.4738)	21.06. 13:12	12.4600	12.4600	n/a	35	
Porsche Automobil Holding SE CFD	2.42%	38/77	21.06. 13:20	close short (Stop 93.82 (Trailing Stop EoP))	21.06. 13:20	93.80	93.82	n/a	35	
Aareal Bank AG CFD	2.38%	61/41	21.06. 13:11	close short (Profit Target 20.5170)	21.06. 13:11	20.4800	20.4400	n/a	35	
Deutsche Boerse CFD	2.25%	12/22	21.06. 13:20	Short (Profit Target 146.65; Stop 147.74 (Trailing Stop EoP))	21.06. 13:20	146.95	146.95	0.00	0	
Japan 225 CFD	1.72%	113/228	21.06. 13:19	Short (Profit Target 28321; Stop 28407 (Trailing Stop EoP))	21.06. 13:21	28351	28341	0.04	35	
Aurubis AG CFD	1.70%	45/54	21.06. 13:13	Long (Stop 76.510 (Trailing Stop EoP))	21.06. 13:13	76.540	76.540	0.00	65	
Qiagen NV CFD	1.62%	29/41	21.06. 13:18	Long (Stop 39.8470 (Trailing Stop EoP))	21.06. 13:18	39.8500	39.8600	N 0.03	100	-
JWSignalMagic(21.06.21 13-20-24)								13		_

Like the WorkspaceBar and the InfoBar you can position the ScreenerBar somewhere in the main window or on the desktop.

17.9.1 The ScreenerBar Columns

Each line of the ScreenerBar contains condensed information of the best found parameter setting of an analyzed security. The signification of the columns is as follows:

Name	The name of the analyzed security.
	If the study violates a trading constraint then



	the name of the security is preceded with an exclamation mark.
Rating	The evaluation of the best found parameter setting.
+/-	Number of win trades/number of losing trades. When using the trading approach <i>Trendsignals</i> , the number of correct signals and wrong signals.
Last Signal	The date of the last signal.
LS Type	The type of the last signal. In case the last signal closed a position, the reason for the signal is displayed in braces, e.g. "(Profit Target 12.88)" or . "(Stop Loss 10.78)".
	In case the last signal opened a position the actual stop and profit targets for this trade are displayed in braces if the study uses these stopping criterions, e.g. "Long (Profit Target 72,32; Stop Loss 62,37)".
	In case the last signal requires a confirmation the confirmation price is displayed, e.g., "Long (trigger: 45.69)".
	In case the last signal was denied because the required confirmation price was not reached, the column reads "Long (filtered)" or "Short (filtered)".
Last Confirmation	The last date with a sentiment value reaching a threshold in the direction of the current trade. Reaching the threshold is a confirmation of the current trade and might be used for increasing the position size or an subsequently entering of the position.
LS Quote	LS = Last Signal The quote at the date of the last signal.
Act Quote	The latest quote.
Act.Trade	The current value of an open position.
Act. Sentiment	The current value of the Meta-Sentimentor.

By clicking a column header the table will be sorted with respect to this column. Clicking again on the same column header will reverse the sorting sequence. Thus, to see the latest signals just click on the column "Last Signal".

Doubleclicking a row of the ScreenerBar will load the corresponding study.

17.9.2 Violations of Trading Constraints

After updating the price data and running a status report for a script it may happen that a study violates its trading constraints, .e.g. because the final has turned into a losing trade that results in an winning/losing trade ratio that is not accepted. In such a case the name of the security in the ScreenerBar is preceded with an exclamation mark so that a violation of trading constraints becomes immediately visible.

Screener	
Name	E
! Beiersdorf CFD	
HerengCement AG CFD	
Merck KGAL CFD	
Rheinmetall VZ C D	
Porsche Automobil Houting SE CFD	
Fresenius Medical Care CFD	
Deutsche Telekom CFD	
Ceconomy AG CFD	

17.9.3 The ScreenerBar Context-Menu

A rightclick on the ScreenerBar opens a context menu.

Screeners					
Name		Rating		+/-	Las
Volkswagen AG Pref CFD		1.38%	:	32/52	21.06
EU Stocks 50 CFD		1 1 69/	15	482	21.06
Rheinmetall VZ CFD	Open			5/48	21.06
Continental AG CFD	Print			1/66	21.06
RWE AG CFD	View	in Browser	NS	7/22	21.06
Henkel AG (Preferred) CFD	Add F	RatingTables		1/46	21.06
Switzerland 20 CFD				5/90	21.06
United Internet AG CFD	Delet	e Page		7/41	21.06
Dialog Semiconductor CFD	Delet	e all Pages		5/23	21.06
JWSignalMagic(21.06.21 13-20-	24)				

The corresponding functionality is as follows:

Open	loads the study (same as doubleclicking the row)
Print	prints the ScreenerBar contents
View in Browser	displays the ScreenerBar contents in a browser (this allows to cut & paste the table, e.g., to send it via email)
Add Rating Tables	More result reports from previous script runs can be loaded. To switch between loaded reports use the tabs at the bottom of the ScreenerBar.
Delete Page	Deletes the displayed report from the ScreenerBar.
Delete all Pages	Deletes all reports from the ScreenerBar.

17.10 Creating a Status-Report

Often one just wants to have an overview of the current status of some securities without applying an optimization. Such an overview is called a *status report*. When creating a status report, the script loads subsequently the quote

files and their respective studies. Each study is evaluated without optimizing it and the result is taken into the status report.

Depending on the personal strategy a once optimized study can be left unchanged for a longer period. After a quote update, it has to be checked whether the new quotes led to new signals. This can be done most easily by just creating a status report.

To create a status report rightclick on the script in the WorkspaceBar and select Create Status Report from the context menu.



NanoTrader will then create the status report on the basis of the script definitions, but ignoring the settings for the stopping criterion, i.e., no optimization is performed.

17.11 Format of Script Files

In some application scenarios it is very convenient to create scripts automatically by an external software. For enabling such a mechanism NanoTrader stores script files as plain ASCII-files carrying the extension .dsb. A dsb-file contains lines of the format:

```
key: value
```

At most one key/value pair per line is accepted. The file may contain empty lines. A line starting with an # is considered a comment.

If a value denotes a pathname it may contain multiple * and ? wildcards. Moreover, using the syntax \$ (<registry key>) a value from the Windows registry may be referenced.

Key	Description	Examples
FilePattern:	Regular expression describing the quote files to be analyzed. Multiple patterns separated by a ; can be provided	\$(QuotesDir)\Dax30*.txt
DefaultDysFile:	dys-file to use in case no dys file with the appropriate label (see below) can be found for a given quote file. After the optimization, a	<pre>\$(DefaultDysDir)\short_buy .dys</pre>



	dys file with the proper label will be created.	
DysLabel:	The label of the dys-files to use.	ShortBuy
LogDir:	Directory for storing log files	\$(ReportsDir)\Dax30
ResultDir:	Directory for storing the result report.	\$(ReportsDir)\Dax30
Algorithm:	Algorithm to be used. Currently only TabuSearch is supported.	TabuSearch
MaxSeconds:	maximal runtime in seconds per underlying. –1 means unlimited.	1200
MaxTries:	maximal number of tries per underlying. –1 means unlimited	-1
UpdateStudies:	Force usage of the defined template study even if a study with the defined study label is already existing (1)1 means use the defined study if existing, otherwise use the template study.	1

17.12 Logfiles

When processing a script, a so-called *activity log* is created. The activity log is being displayed in the ScriptViewer. Moreover, a result report is created that is automatically displayed in the ScreenerBar after the script run has terminated.

The filename of activity logs is of the format: <Dyslabel>_log_<date time>.log.

Result report files are named:

<Dyslabel>_result_<date time>.log

18 Manually defined Sentimentors

18.1 Areas of Application

A sentimentor can be created manually by defining fixed sentiment values for time periods. Typically, a manually defined sentimentor is used in conjunction with the Meta Sentimentor to bias the sentiment in certain periods and thus to influence the meta sentiments as well as the resulting signals.

The definition of such a *Manual Sentimentor* is extremely valuable whenever non-formal and/or fuzzy knowledge should be taken into account for an analysis, e.g.,

- exchanging companies in an important index
- uncertainty before the next FED meeting
- Making ones intuition explicit. This is a very comfortable way to create and communicate "complete" studies. Moreover, making ones intuition and feelings explicit allows to learn and improve by recapitulating the decisions of the past.

- empirical peaks preceding important announcements (e.g., raising quotes at the customer fairs of SAP – this is normally the time to announce strategic decisions)
- retrospective incorporation of important company announcements.

Here is an example for the latter point:

An unexpected merger announcement may lead to a dramatic change in the quotes as exemplified with the announced merger and subsequent merger cancellation of Deutsche Bank and Dresdner Bank:



By defining a Manual Sentimentor the announcements can be captured. To achieve this the sentiment values are set to 100 at the announcement of the merger and to 0 at the cancellation announcement. By incorporating this Manual Sentimentor into future Deutsche Bank analyses, it will be ensured that the system is invested in the "correct" direction with respect to the announcements. This releases the optimization from trying to find parameters that would take advantage of these drastic quote changes. This is important as there is no realistic chance for repeating such a quote pattern in the future.

18.2 Editing a Manual Sentimentor

The editor for Manual Sentimentors can be started by rightclicking on a Manual Sentimentor and selecting Edit from the context menu.





The editor looks as follows:

News, Dautache D			
Name: Deutsche E	ank		
Format			
Each line conta	is the sentiment value for the specified time range.		
The following fo	mat is expected:		
dd.mm.yy - dd.m	n.yy : value comment URL		
i.e. a valid line l	oks like this		
12.12.99 - 13.01	00 : 25 Euro in a bad shape http://www.rec.com/e	uro.pdf	
2.12.99 - 28.12.99:	100 Merger Announcement Dresdner Bank	Υ	-
17.03.00 - 15.03.00:	U Merger Failed	T	
			~
4			- F

A manually defined sentimentor is a text consisting of lines with the following format:

dd.mm.yy[yy] - dd.mm.yy[yy]: sentiment comment

sentiment represents the sentiment value for the specified period, thus it must be an integer between 0 and 100. The specification of a comment is optional.

Obey the – and : characters when editing a line. Spaces and tabs may be used to format a line.

If time periods are overlapping, the last value for a multiply covered date is taken.

For all are uncovered dates, NanoTrader uses the neutral sentiment (50).

The comment will be displayed by NanoTrader when the Manual Sentimentor is used in a study and the mouse points to a period with a specified sentiment.



18.3 Creating a New Manual Sentimentor

To create a new Manual Sentimentor start the Manual Sentimentor Editor by rightclicking on the entry Manual Sentimentors in the WorkspaceBar.

V				
	Manual Sentimentor	s		_
	🖞 Deutsche Bank	Ð	New	
	🖞 FED-Fear	5	Defrech	45
	🖞 SAP-Sapphire	G	Kellesh	

18.4 Creating a new Manual Sentimentor from an existing

If you want to change and save an existing Manual Sentimentor with a new name proceed as follows:

- 1. Load the Manual Sentimentor in the Manual Sentimentor Editor.
- 2. Overwrite the name of the Manual Sentimentor with the new name.
- 3. Change the sentiment specifications at your wish.
- 4. Save the Manual Sentimentor.



18.5 Deleting a Manual Sentimentor

In the WorkspaceBar, rightclick on the Manual Sentimentor to be deleted and choose Delete from the context menu.



18.6 Adding a Manual Sentimentor to a Study

As all other sentimentors, the Manual Sentimentors can be added to a study using the Add Sentimentor dialogL

Type to search	>>	Crossing MA Volume Viewer Exp. Moving Average Time Price Opportunities Key Price Levels Volume Profile LiveStatistics Express\PrevDayClose Bollinger Bands Kaufmann's AMA Meta Sentimentor Express\EMA Trailing Stop Trailing Stop EoP MACD	
Display in MasterChart Display as subwindow in MasterChart windo Insert as: Sentimentor Filter	ow Stop	Tactic Close	

18.7 Manual Sentimentors used as Filters

Sometimes it is desired to exclude certain periods in time from trading, e.g., the opening phase of a trading session or the five minutes before and after import statistics are announced. In these periods all signals should be suppressed.

To achieve this Manual Sentimentors can be created in the Timed Exits and Filters section of the WorkspaceBar:





Instead of sentiment values the symbolic names *FLAT* and *Block* are used:

Timed Exit & Filter - Editor	
Name: Intraday	
Format Each line contains the sentiment value for the specified time range. The following format is expected: DD.MM.YY HH:MM - HH:MM : <block flat="" or="">Comment Omit the date for recurring actions, e.g:</block>	
9:00 - 9:30 : Block Opening Chaos	
08:00 - 09:30 : BLOCK. Opening Blocker 15:15 - 15:45 : FLAT US Blocker 13:00 - 23:59 : FLAT Session End	*
	Ŧ
4 P	
Save Cancel	

FLAT ensures that an open position is closed and that new entry signals are suppressed.

BLOCK is less restrictive in that open positions are left untouched until an exit signal occurs, but entry signals are suppressed.

In case the periods are repetitive, as in the example above, no date needs to be specified. If the period is bound to a specific date then the complete date and time need to be specified.

The MasterChart visualizes BLOCK and FLAT periods with light blue and blue:





The FLAT sentiment is triggered precisely at the time it is defined to start, even if it is in the midst of a period, given the Evaluator setting for "Sentimentor Exit Signal" is set to "Close same bar", i.e., in this case it is *not* waited for the end of the period. This setting captures the most important case of closing at a specific time, e.g., shortly before an important news release, no matter what the chart aggregation is set to.

If in the Evaluator the "Sentimentor Exit Signal" is set to "Open next bar" the FLAT command is only executed with the open of the next bar of the MasterChart.

18.8 The "Flat" and "Block" Filters - Exiting at the end of the day

An easy alternative to the definition of filters in files are the built-in filters called "Flat" and "Block":

Type to search	2		×	+
Indicators Indicators in Express Manual Stops Timed Exits & Filters Flat Block Safety Net Discard Price Data Intraday WHS Store	>>	Crossing MA Volume Viewer Exp. Moving Average Time Price Opportunities Key Price Levels Volume Profile LiveStatistics Express\PrevDayClose Bollinger Bands Kaufmann's AMA Meta Sentimentor Express\EMA Trailing Stop Trailing Stop EoP MACD		
 Display in MasterChart Display as subwindow in MasterChart 	window			



They provide an easy mechanism to define one time interval where they emit the Flat or Block sentiment:

DesignerBar	оооссологооссолого ф .
🗞 - 🖌 🗙 🔅 🔀	📇 🛹 \$+ 🐏 🔳 🖬
Trading	
Brackets	
Tactic Buttons	
✓ Filters	
 ✓ Filters ✓ Flat 	18:00 - 23:59
 ✓ Filters ✓ Flat ✓ Block 	18:00 - 23:59 13:00 - 13:59
 Filters Flat Block Indicators 	18:00 - 23:59 13:00 - 13:59

They can be activated or deactivated by checkmarking them, i.e., if you decide to keep your position open even after 18:00 you would just uncheck the Flat filter.

See the comments in the previous section on how the starting time is converted into action when the beginning of a FLAT period falls in the middle of a charted period.

18.9 Creating a Manual Sentimentor with an External Tool

Manual Sentimentors can also be used to incorporate results of other programs or specific indicators into NanoTrader.

To enable this process, Manual Sentimentors are stored as plain ASCII files with the suffix .sent. Such a file must be stored in the ManSentisDir, which is by default a subdirectory of the installations directory.

19 Parameters of the Sentimentors

19.1 Sentimentors Based on Technical Indicators

Each sentimentor relying on a technical indicator will be described by its computation, its parameters, and the generation of the sentiment values. The generation of a sentiment value is always discussed with respect to a specific date *t*. In order to keep the description easy, we will not mention this fact explicitly any more.

19.1.1 ADX

The ADX is a trend indicator that merely states the strength of a trend, but not the trend direction. Therefore, the ADX can only be used as a *blocker*, i.e., in case the ADX shows the existence of a trend, enter signals in both directions are passed, otherwise enter signals are blocked.

Computation:

plusDM[i] = max(high[i] – high[i-1], 0)

minusDM = max(low[i-1] - low[i], 0)

trueRange[I] = max (close[i-1], high[i]) - min (close[i-1], low[i]);

sumPlusDM = sum over the last Span plusDM-values

sumMinusDM = sum over the last *Span* minusDM-values

sumTrueRange = sum over the last Span trueRange -values

plusDI = 100 * sumPlusDM / sumTrueRange

minusDI = 100 * sumMinusDM / sumTrueRange

averagePlusDI = MA over the last Span DM periods of plusDI

averageMinusDI = MA over the last Span DM periods of minusDI

DMI = 100 * abs (averagePlusDI – averageMinusDI) / (averagePlusDI + averageMinusDI)

ADX = MA over DMI with length Span ADX

Parameter:

Span: number of periods to consider for computing the DMI

DM Span: speed of the MA for averagePlusDI and averageMinusDI

ADX Span: speed of the MA for the ADX

Trend Threshold: values above the threshold indicate a trend

Interpretation:

ADX >= Trend Threshold: enter signals are passed

ADX < Trend Threshold: enter signals are blocked

19.1.2 Aroon

Computation:

i_{max} = number of periods since the maximum of the last *Span* periods occurred

i_{min} = number of periods since the minimum of the last *Span* periods occurred

bandMax = 100 * (*Span* - i_{max}) / *Span* bandMin = 100 * (*Span* - i_{min}) / *Span*

Parameter:

Span: number of periods to consider

Threshold: Aroon-threshold

Interpretation:

bandMax crosses the threshold from below and bandMin < threshold: sentiment value = 100

bandMax crosses the threshold from below and bandMin crosses the threshold from above: sentiment value = 100

bandMin crosses the threshold from above and bandMax > threshold: sentiment value = 75



bandMin < threshold and bandMax > threshold: sentiment value = 75

bandMin crosses the threshold from above and bandMax < threshold: sentiment value = 65

bandMax crosses the threshold from above and bandMin < threshold: sentiment value = 35

bandMax < threshold and bandMin > threshold: sentiment value = 25

bandMax crosses the threshold from above and bandMin > threshold: sentiment value = 25

bandMin crosses the threshold from below and bandMax crosses the threshold from above: sentiment value = 0

bandMin crosses the threshold from below and bandMax < threshold: sentiment value = 0

otherwise: sentiment value = 50

19.1.3 ATR

The ATR is an indicator which measures the size of a movement in ticks. As such it indicates not a direction, but the significance of a movement. Hence it can only be used as a *Blocker* (similar to the ADX) that accepts signals, given the movement is sufficiently strong, or rejects them otherwise.

Therefore the ATR is very well suited for filtering signals in sideway ranges.

Computation:

The Average True Range is a moving average on the True Range which is defined as the largest difference of

- today's high minus today's low
- today's high minus yesterdays close
- yesterdays close minus toady's low

Parameter:

Span: number of periods for computing the MA on the True Range

Threshold: values above the threshold (in ticks) indicate a strong movement

Interpretation:

ATR >= Threshold: enter signals are passed

ATR < Threshold: enter signals are blocked

19.1.4 Bollinger Bands

Computation:

MA = moving average over the last Span periods

s = standard deviation over the last *Span* periods

bandUp = close_t + Factor (1/10 StdDev) * s

bandDown = closet - Factor (1/10 StdDev) * s

Parameter:

Span: number of periods to consider for the moving average and the standard deviation

Factor (1/10 StdDev): factor for the width of the bands, measured in 1/10 of the standard deviation

Interpretation:

close crosses bandUp from below: sentiment value = 100

close > bandUp: sentiment value = 75

close crosses bandDown from below: sentiment value = 65

close crosses bandUp from above: sentiment value = 35

close < bandDown: sentiment value = 25

close crosses bandDown from above: sentiment value = 0

otherwise: sentiment value = 50

19.1.5 CCI – Channel Commodity Index

Computation:

1.) **X**t= (Hight + Lowt + Closet) / 3 the so-called "significant quote"

2.) MA(X) = Moving Average of the significant quote

3.) $sX_t = (abs (X_t - MA(X)_t) + abs(X_{t-1} - MA(X)_t) + ... + + abs(X_{t-n+1} - MA(X)_t))/n$ standard deviation of the significant quote

4.) CCIt = $(X_t - MA(X)_t) / (0.015 * sX_t)$

Parameter:

Span: Parameter *n* used in the computation

Threshold Up-Trend: CCI-threshold indicating an upward trend

Threshold Down-Trend: CCI-threshold indicating an downward trend

Interpretation:

Interpretation scheme "Two Zone" with a modified default setting:



Interpretation for Two Zones	×
1. 3.	
2. 4.	
	6.
5. 7.	
Event:	Sentiment Series starting at event period, e.g. 100;90
1. Entering Upper Zone:	100;
2. Staying in Upper Zone:	75
3. Leaving Upper Zone:	40;
4. Staying between Zones:	50
5. Entering Lower Zone:	0;
6.Staying in Lower Zone:	25
7. Leaving Lower Zone:	60;
Template	
*	Select
Remove	Save
ОК	Cancel

19.1.6 Channel Breakout

Computation:

First of all, a moving average (MA) of the close quotes is computed. Then the highest and lowest quotes per period are determined

Parameter:

Smoothness Span: speed of the MA

Period Length: length of the period for computing the highs and lows.

New Highs Long: number of required subsequent highs until a positive breakout is stated

New Lows Short: number of required subsequent lows until a negative breakout is stated.

Interpretation:

positive breakout: sentiment = 100

negative Breakout: sentiment = 0

else:

The sentiment is determined by the relative position of the smoothed close quote with respect to the periods high and low mapped onto the sentiment interval [35-65], i.e.

sentiment $t = 35 + (MA_t - period low_t) / (period high_t - period low_t) * 30$

19.1.7 Crossing MA

Computation:

Two moving averages with a different speed are computed from the close prices.

Parameter:

Fast MA Span: speed of the fast MA

Slow MA Span: speed of the slow MA

Interpretation:

slow MA > fast MA: sentiment value = 35

slow MA <= fast MA: sentiment value = 65

fast MA crosses the slow MA from below: sentiment value = 100

fast MA crosses the slow MA from above: sentiment value = 0

19.1.8 Directional Ind. (+/-DI)

Computation:

plusDM[i] = max(high[i] – high[I-1], 0) minusDM = max(low[I] – low[I-1], 0) trueRange[I] = max (close[i-1], high[i]) - min (close[i-1], low[i]); sumPlusDM = sum over the last *Span* plusDM-values sumMinusDM = sum over the last *Span* minusDM-values sumTrueRange = sum over the last *Span* trueRange -values plusDI = 100 * sumPlusDM / sumTrueRange minusDI = 100 * sumMinusDM / sumTrueRange averagePlusDI = MA over the last *Span* periods of plusDI averageMinusDI = MA over the last *Span* periods of minusDI **Parameter:** Span: number of periods to consider MA Span: speed of the MA

Interpretation:

averagePlusDI crosses averageMinusDI from below: sentiment value = 100 averagePlusDI crosses averageMinusDI from above: sentiment value = 0 averagePlusDI > averageMinusDI: sentiment value = 65 averagePlusDI < averageMinusDI: sentiment value = 35 otherwise: sentiment value = 50

19.1.9 DMI – Dynamic Momentum Index

Computation:

s = standard deviation over the last *Deviation Span* periods

S = MA over the last *Smoothness Span* periods of s

T = INT (14 * S / s)

RS = (sum of the ascending close-values of the last T periods) / (sum of the descending close-values of the last T periods)

RSI = 100 - (100 / (1 + RS))

Parameter:

Deviation Span: number of periods to consider for the standard deviation

Smoothness Span: speed of the MA

Overbought Threshold: threshold for the overbought region

Oversold Threshold: threshold for the oversold region

Interpretation:

DSI-curve leaves the overbought region => sentiment value = 0

DSI-curve leaves the oversold region => sentiment value = 100.

Otherwise: sentiment value = 50

19.1.10 DSS – Double Smoothed Stochastik

Computation:

$$low[i] = \min_{j=0..Span-1} close[j]$$

$$high[i] = \max_{j=0..Span-1} close[j]$$

$$DSS = 100 \cdot \frac{XMA_{1.EMA-Span}(XMA_{2.EMA-Span}(close-low))}{XMA_{1.EMA-Span}(XMA_{2.EMA-Span}(high-low))}$$

Parameter:

Span: number of periods to consider for the computation of the maximums and minimums

1.EMA-Span: speed of the first EMA

2.EMA-Span: speed of the second EMA

Threshold Up: threshold for the overbought region

Threshold Down: threshold for the oversold region

Interpretation:

DSS-curve leaves the overbought region => sentiment value = 0

DSS-curve leaves the oversold region => sentiment value = 100.

Otherwise: sentiment value = 50

Financial Performance Technology

19.1.11 Exponential Moving Average

Computation:

 $EMA[t] = EMA[t-1] + (SF \cdot (Ct - EMA[t-1]))$

with

SF = 2/(span + 1)

Parameter:

Span: number of periods for computing SF..

Interpretation:

close crosses average from below: sentiment value = 100

close crosses average from above: sentiment value = 0

close > average: sentiment value = 65

close < average: sentiment value = 35

Otherwise: sentiment value = 50

19.1.12 Heikin Ashi

The Heikin Ashi is similar to a candle stick chart, but it uses modified data for open/high/low/close for each candle. By this modification the traders can identify a trend more easily by means of visual inspection. Bull candles without lower shadows signal a strong uptrend, while Bull candles with no upper shadow signal a downtrend.

Computation:

HA Close = (Open+High+Low+Close)/4 HA Open = [HA Open (previous bar) + HA Close (previous bar)]/2 HA High = max (High, HA Open, HA Close) HA Low = min (Low, HA Open, HA Close)

Parameter:

There are no parameters.

Interpretation:

The sentiment is fixed to be 50. Heikin Ashi is used for discretionary trading only.

19.1.13 KAMA - Kaufmann's Adaptive Moving Average

Computation:

 $efratid[i] = \frac{|close[i] - close[i - span]|}{\sum_{j=i-span+1}^{i} |close[j] - close[j-1]|}$

Financial Performance Technology

$$constFASTEST = \frac{2}{2+1}$$

$$const SLOWEST = \frac{2}{30+1}$$

 $smoothed[i] = (efratid[i]*(FASTEST - SLOWEST) + SLOWEST)^{2}$

 $KAMA[i] = KAMA[i-1] + smoothed[i] \cdot (close[i] - KAMA[i-1])$

Parameter:

Span: number of periods to consider

Interpretation:

close crosses KAMA from below: sentiment value = 100

close crosses KAMA from above: sentiment value = 0

close > KAMA: sentiment value = 65

close < KAMA: sentiment value = 35

Otherwise: sentiment value = 50

19.1.14 Linear Regression

Computation:

A linear fit is calculated through the last *Smoothness Span* periods. The value of this linear regression is calculated for the current day and stored in %regression.

A moving average over the last *Span* periods is also calculated and stored in %Smoothed.

Parameter:

Span: number of periods for the linear regression

Smoothness Span: speed of the MA

Interpretation:

%smoothed crosses %regression from below: sentiment value = 100

%smoothed crosses %regression from above: sentiment value = 0

%smoothed > %regression: sentiment value = 65

%smoothed < %regression: sentiment value = 35

Otherwise: sentiment value = 50

19.1.15 Local Highs & Lows

Computation:

A MA from the close quotes is computed.

Parameter:

Smoothness Span: Speed of the MA

Span Left: examination range "before" an extreme point

Span Right: examination range "after" an extreme point

Interpretation:

There exists a *local high* at the date *t*, if for "span left" periods before *t* the smoothed close quotes are ascending and for "span right" periods after *t* the smoothed close quotes are descending. A *local low* is defined analogously.

local high at date t: sentiment value = 0 at date t + "Span Right"

local low at date *t*: sentiment value = 100 at date *t* + "Span Right"

otherwise: sentiment value = 0

19.1.16 MACD

Computation:

Computation of the difference of two exponential moving averages (EMA) of the close quotes as well as an EMA on that difference. The latter EMA is called the "trigger".

Parameter:

EMA diff fast: speed of the fast EMA used for computing the difference

EMA diff slow: speed of the slow EMA used for computing the difference

EMA: speed of the "trigger" EMA

Interpretation:

EMA diff > Trigger: sentiment value = 65

EMA diff < Trigger: sentiment value = 35

EMA diff crosses Trigger from below: Sentiment value = 100

EMA diff crosses Trigger from above: Sentiment value = 0

19.1.17 MACD-Histogram

Computation:

Like MACD, but finally the difference between "EMA diff" and the "Trigger" is computed.

Parameter:

EMA diff fast: see MACD

EMA diff slow: see MACD

EMA: see MACD

Span Left: see "Local Highs & Lows"

Span Right: see "Local Highs & Lows"

Interpretation:

The sentiment values are derived from the extreme points of the computed difference applying the scheme as described for "Local Highs & Lows".



19.1.18 Momentum

Computation:

First of all a MA on the close quotes is computed. The Momentum for date t is then computed as

 $MOM_t = ((MA_t / MA_{t-1}) - 1) * 100$

Parameter:

Smoothness Span: speed of the MA

Span Left: see "Local Highs & Lows"

Span Right: see "Local Highs & Lows"

Interpretation:

The sentiment values are derived from the extreme points of the computed Momentum applying the scheme as described for "Local Highs & Lows".

19.1.19 Moving Average

Computation:

$$averag[i] = \frac{1}{Span} \cdot \sum_{i=i-Span+1}^{i} close[i]$$

Parameter:

Span: number of periods.

Note: Wherever a moving average is to be computed, also in other sentimentors, a *negative* value for the span might be provided. This results in applying an exponential moving average. Through this scheme, the optimization engine might switch silently from linear to exponential smoothing.

Interpretation:

close crosses average from below: sentiment value = 100

close crosses average from above: sentiment value = 0

close > average: sentiment value = 65

close < average: sentiment value = 35

Otherwise: sentiment value = 50

19.1.20 On-Balance-Volume

Computation:

OBV₀ = 0 If Close today > Close yesterday: OBV today = OBV yesterday + volume today If Close today < Close yesterday: OBV today = OBV yesterday - volume today

else

OBV today = yesterday

On the OBV a MA is computed.



Parameter:

Smoothness Span: speed of the MA of OBV

Span Left: see "Local Highs & Lows"

Span Right: see "Local Highs & Lows"

Interpretation:

The sentiment values are derived from the extreme points of the computed MA applying the scheme as described for "Local Highs & Lows".

19.1.21 Orderbook

Computation:

Based on the current order book the following is calculated:

AskSurplus = (total ask volume - total bid volume) *100% / total volume; i.e. the surplus of Ask volume in percentages of the total volume.

The display below the legend is updated with every orderbook change. The chart updates only when a new tick arrives and shows the ask surplus at this time.

Typically this sentimentor is used for studies working in Tick aggregation.

Note: As there is no internal saving of the orderbook data the historic data is lost whenever the chart aggregation is changed.

Parameter:

MA-Span: speed of the MA of AskSurplus

Max Book Depth: Number of orderbook entries to consider, e.g., if only the 3 best Bid/Ask offers should be evaluated then set this parameter to 3.

Threshold Buy: Buy-Threshold

Threshold Sell: Sell-Threshold

Interpretation: Scheme "Two Thresholds"

19.1.22 Parabolic SAR

Computation: Please see "Parabolic Stop".

Interpretation:

close crosses Parabolic SAR from below: sentiment value = 100

close crosses Parabolic SAR from above: sentiment value = 0

close > Parabolic SAR: sentiment value = 65

close < Parabolic SAR: sentiment value = 35

Otherwise: sentiment value = 50
19.1.23 PFE – Polarized Fractal Efficiency

Computation:

$$PFE_unsigned[i] = \frac{\sqrt{(close[i] - close[i - span])^2 + span^2}}{\sum_{j=i-span+1}^i \sqrt{(close[j] - close[j + 1])^2 + 1}}$$
$$PFE[i] = \begin{cases} PFE_unsigned[i] & for close[i] > close[i - span] \\ -PFE_unsigned[i] & otherwise \end{cases}$$

smoothed = MA over the last Smoothness Span periods of PFE

Parameter:

Span: number of periods considered in the PFE

Threshold Buy: threshold for the overbought region

Threshold Sell: threshold for the oversold region

Smoothness Span: speed of the MA

Interpretation:

smoothed crosses *Threshold Buy* from below: sentiment value = 100 smoothed > *Threshold Buy*: sentiment value = 75 smoothed crosses *Threshold Buy* from above: sentiment value = 40 smoothed crosses *Threshold Sell* from above: sentiment value = 60 smoothed > *Threshold Sell*: sentiment value = 25 smoothed crosses *Threshold Sell* from below: sentiment value = 0 Otherwise: sentiment value = 50

19.1.24 Pivot-Points

The Pivot Points are computed based on the previous' day trading range. Quite often they mark significant support/resistance levels for the actual trading day.



The visualization displays the support/resistance lines at the level of the Pivot Points. Moreover, a moving average of the close is displayed. This MA is used for generating the signals. In case the span of the MA is set to 0, the MA is not drawn and the close prices of the bars are taken for the signal generation.

Computation:

Pivot = (PrevDayHigh + PrevDayLow + PrevDayClose) / 3

Resist 3 = 2 * (P – PDL) + PDH Resist 2 = P + (PDH – PDL) Resist 1 = (2 * P) – PDL Support 1 = (2 * P) – PDH Support 2 = P – (PDH – PDL) Support 3 = 2 * (P – PDH) + PDL

Parameter:

Span: speed of the MA

Support-Delta: absolute price value for defining the support zone (0 = no support zone)

Resist-Delta: absolute price value for defining the resist zone (0 = no resist zone)

Interpretation: Scheme "Support/Resistance", whereby the MA is used for generating the signals.

19.1.25 Point & Figure

The Point & Figure technique transforms the given price data in a specific way in order to just show "significant" price moves.

Parameter:

Box-Size: Defines the size of a box. In case this value is given as a positive number it is interpreted as an absolute price value. A negative number is interpreted as a percentage.

Reversal Amount: Required minimum change in boxes in order to signal a trend reversal.

HiLo; Close: Defines if the boxes are calculated based on the High/Lows or on the Close price. (Choose 0 for High/Low; choose 1 for Close).

Interpretation: The interpretation can be defined through the "Point & Figure" scheme:



Interpretation for Point & Fig	ure ×
	4.
Event:	Sentiment Series starting at event period, e.g. 100;90
1. Start of X column:	65;
2. Continuation of X column:	65
3. Double-Top:	100;
4. Start of O column:	35;
5. Continuation of O column:	35
6. Double-Bottom:	0;
Template	
•	Select
Remove	Save
ОК	Cancel

19.1.26 Renko

The Renko technique transforms the given price data in a specific way in order to just show "significant" price moves.

Parameter:

Box-Size: Defines the size of a box. In case this value is given as a negative number it is interpreted as an absolute price value. A positive number is interpreted as a percentage.

Reversal Amount: Required minimum change in boxes in order to signal a trend reversal.

Interpretation: The interpretation can be defined through the "Renko" scheme:



Interpretation für Renko	¢
3. 4.	
Sentiment Series starting at Event: event period, e.g. 100:90	
1. Start of Upward-Trend: 100;	
2. Continuation of Upward-Trend: 65	
3. Start of Downward-Trend: 0;	
4. Continuation of Downward-Trend: 35;	
Template	
- Select	
Remove Save	
OK Cancel	

19.1.27 RSI

Computation:

RS = (sum of the quote gains of the last "Span" periods) / (sum of the quote losses of the last "Span" periods)

RSI = 100 - (100 / (1 + RS))

Parameter:

Span: number of periods to consider

Overbought Threshold: threshold for the overbought region

Oversold Threshold: threshold for the oversold region

Interpretation:

RSI-curve leaves the overbought region => sentiment value = 0

RSI-curve leaves the oversold region => sentiment value = 100.

Otherwise: sentiment value = 50

19.1.28 RSI-smoothed

Computation:

RS = (sum of the quote gains of the last "Span" periods) / (sum of the quote losses of the last "Span" periods)

RSI = 100 - (100 / (1 + RS))

RSI-smoothed = MA over the last Smoothness Span periods of RSI

Parameter:

Span: number of periods to consider

Overbought Threshold: threshold for the overbought region



Oversold Threshold: threshold for the oversold region

Smoothness Span: speed of the MA

Interpretation:

RSI-smoothed-curve leaves the overbought region => sentiment value = 0

RSI- smoothed -curve leaves the oversold region => sentiment value = 100.

Otherwise: sentiment value = 50

19.1.29 Slow Stochastic

Computation:

%K = (close - low_{Span}) * 100 / (high_{Span} - low_{Span})

%Fast = MA over the last Span %D periods of %K

%Slow = MA over the last Span %D periods of %Fast

where low_{Span} is the minimum over close of the last *Span periods and high*_{Span} is the maximum over close of the last *Span periods*.

Parameter:

Span %K: number of periods considered to calculate %K

Span %D: speed of MA for %Slow and %Fast

Overbought Threshold: threshold for the overbought region

Oversold Threshold: threshold for the oversold region

Interpretation:

%D leaves overbought region => sentiment value = 0

%D leaves oversold region => sentiment value = 100

Otherwise: Sentiment value = 50

19.1.30 Stochastic

Computation:

%K = (close – low_{Span}) * 100 / (high_{Span} – low_{Span})

%D = MA of %K

with low_{Span} denotes the low of the last *Span* periods, and high_{Span} denotes the high of the last *Span* periods.

Parameter:

Span %K: span in periods for the %K-computation

Span %D: speed of the %D MA

Overbought Threshold: threshold for the overbought region

Oversold Threshold: threshold for the oversold region

Interpretation:

%D leaves overbought region => sentiment value = 0

%D leaves oversold region => sentiment value = 100

Otherwise: Sentiment value = 50

19.1.31 Study

The <u>Section "Cascading Studies"</u> for a complete description.

19.1.32 SuperTrend

This indicator was developed by Olivier Seban.

The main idea is to use the smoothed Average True Range in a trailing manner, i.e., in an uptrend the SuperTrend value can only increase, in a downtrend the SuperTrend value can only decrease.

Whenever the closing price of a period crosses the SuperTrend, a trend reversal is established.

Computation:

avgATR = MA of the ATR for the last Span periods

longStop = max (previous longStop, (high + low) / 2 - avgATR * ATR_factor)

shortStop = min (previous shortStop, (high + low) / 2 + avgATR * ATR_factor)

If in Uptrend then SuperTrend = longStop; else SuperTrend = shortStop. If close crosses the SuperTrend the trend mode is reversed.

Parameter:

Span: span in periods for the MA computation of the ATR

ATR factor: multiplier for the avgATR offset.

Interpretation:

close >= SuperTrend: sentiment value = 65

close < SuperTrend: sentiment value = 35</pre>

close crosses the SuperTrend from below: sentiment value = 100

close crosses the SuperTrend from above: sentiment value = 0

19.1.33 Williams' Variable Accumulation Distribution (WVAD)

Computation:

The WAVD is calculated iteratively:

 $WVAD[i] = WVAD[i-1] + \frac{close[i] - open[i]}{high[i] - low[i]} \cdot volume[i]$

After that two moving averages are calculated over the WAVD:

%Fast = MA over the last Fast MA-Span periods of WVAD

%Slow = MA over the last *Slow MA-Span periods of WVAD*

Parameter:

Fast MA-Span: speed of the fast MA

Slow MA-Span: speed of the slow MA



Interpretation:

%Slow crosses %Fast from below: Sentiment value = 100

%Slow crosses %Fast from above: Sentiment value = 0

%Slow > %Fast: Sentiment value = 65

%Slow < %Fast: Sentiment value = 35

Otherwise: Sentiment value = 50

19.1.34 Williams %R

Computation:

%R = -(high_{Span} - close) * 100 / (high_{Span} - low_{Span})

with lowspan denotes the low of the last *Span* periods, and highspan denotes the high of the last *Span* periods.

Note: in order to let the R display the overbought region at the top of its chart and the oversold region at the bottom the R is multiplied with -1.

Parameter:

Span: span in periods for the %R-computation

Overbought Threshold: threshold for the overbought region

Oversold Threshold: threshold for the oversold region

Interpretation:

curve leaves overbought region => sentiment value = 0

curve leaves oversold region => sentiment value = 100

Otherwise: Sentiment value = 50

19.1.35 Volume

The Volume sentimentor can only be used as a *blocker*, i.e., in case the Volume exceeds a threshold, enter signals in both directions are passed, otherwise enter signals are blocked.

Computation:

No computation

Parameter:

Threshold: necessary volume to let enter signals pass

Vol * 10^x: determines the multiplier for the Threshold (see below).

Span: speed of the volume MA.

As the volume may range from several contracts to a daily volume of several million shares, the given Threshold is multiplied with a power of 10 – this allows to work with a small number that fits well into the Designer-table. The following table gives some examples:

0	1
1	10
2	100
3	1.000
4	10.000
5	100.000
6	1.000.000

Thus:

Threshold Para	x	Resulting Threshold
25	1	25
3	3	3.000
500	3	500.000
2	6	2.000.000

Interpretation:

MA of Volume >= Threshold: enter signals are passed MA of Volume < Threshold: enter signals are blocked

19.2 Candle Stick

19.2.1 Visualization of Candle Stick patterns

Most of the candle stick patterns are need to appear in a certain trend in order to be significant. Therefore, the Candle Stick sentimentor works in two phases: It starts with determining trends using the CCI indicator, followed by searching for certain candle stick patterns. A pattern is only taken into account if it occurs within the required trend.

NanoTrader uses some specific visualization techniques to make this twophase approach as transparent as possible. Let's have a look at the following screenshot:

First of all, above the date axis red and green lines are displayed to indicate the current trend. A green line is used for up trends, a red line for down trends. If no line is displayed, we have a trendless phase.

When moving the mouse over the candle sticks, NanoTrader displays all recognized patterns in popup windows. In case a pattern is not taken into account for generating sentiments, the popup also displays the reason for this

exclusion. This might be, because the pattern does not lie within the correct trend:



... or because the pattern has been deactivated in the sentimentor:



To see at a glance where valid patterns have been recognized that have been taken into account for generating sentiments, NanoTrader draws little blue squares at the of the value scale: (Note that recognizing a pattern does not automatically lead to a signal, e.g., if the system is already invested in the corresponding direction.)





19.2.2 Deactivation of the Trend Component

To deactivate the trend component of the Candle Stick sentimentor, simply fix the first parameter to a value of 1:



19.2.3 The Recognized Candle Stick Patterns

All of the recognized patterns indicate a trend reversal. The sentiment value is generated for the day a pattern has been completed.

If no pattern is completed for a given day, the sentiment value is set to neutral (50) at that day.

Name	Pattern	Sentiment
Morning Star	[₩] ₽ _₽ ¢	100
Evening Star	,¢ [¢] ♥	0



-

Three Soldiers	^۱ ۵۵	100
Three Crows	## # #	0
Bullish Engulfing	[₩] ŧĊ	75
Bearish Engulfing	µ¢∎	25
Outside Up	[⋈] ≢Ů	100
Outside Down	,,¢∎∎	0

The sentiments can be configured in the sentiment editor:

Interpretation forr CandleStick	x-Patterns ×
Pattem:	Sentiment Series starting at pattern completion, e.g. 0;0
Moming Star:	100;
Evening Star:	0;
Three Soldiers:	100;
Three Crows:	0;
Bullish Engulfing:	75;
Bearish Engulfing:	25;
Outside Up:	100;
Outside Down:	0;
Template	
	- Select
Remove	Save
ОК	Cancel

19.2.4 The Parameters of the Candle Stick Sentimentor

The first three parameters are the same as described for the CCI-Sentimentor. If the "Threshold Up-Trend" is exceed, an up trend is stated and vice versa for the "Threshold Down-Trend".

The other parameters refer respectively to candle stick patterns. If a parameter is set to 1, the corresponding pattern taken searched for. If set to zero, the corresponding pattern is ignored.

Parameter:

Span: Parameter *n* used in the computation. A value of 1 deactivates the trend component such that a pattern is stated independently of the current trend.

Threshold Up-Trend: CCI-threshold indicating an upward trend

Threshold Down-Trend: CCI-threshold indicating an downward trend

search = 1; ignore = 0
search = 1; ignore = 0

19.3 Support/Resistance-Sentimentor

19.3.1 The Computation of Support/Resistance-Lines

The support/resistance sentimentor computes price levels that can be interpreted as support/resistance lines. If the quotes approach such a price level from above the level is interpreted as a support line and it is assumed that the prices do not fall below this level. Analogously such a price level is interpreted as a resistance line if the prices approach this level from below.

Before a price level is interpreted as a support/resistance line the prices have to show several turnarounds "near" this level. "Near" means the price must be contained inside an *x*-percent interval of the price level in question. A turnaround is defined as two ascending prices followed by two descending prices. The low or top of a turnaround has be near the price level. This method of computing support/resistance lines ensures that the interpretation is always independent from the future.

The following example shows the computation of a support/resistance line, where "near" is defined as a 2% interval and 5 turnarounds are required before the line is established and taken into account for interpretation:



The numbering shows the five required turnarounds near the support/resistance line. (Note that a turnaround is finished two periods after the respective high or low. This is reflected in the graphic.)

As long as the line does not show enough tests, it is drawn as a dashed line. When it becomes a valid line after the fifth test its support region is visualized with green and the resistance region with red.

In order to have some meaningful interpretation of the support/resistance lines for a security it is necessary that their respective support/resistance regions do not overlap. If this is the case for a pair of lines that one with the most hits is taken. If both lines have the same number of hits the line is taken that reaches the minimal required hits earlier.

19.3.2 Using the Support/Resistance-Sentimentors in Constant Studies

By construction of the support/resistance lines it may happen that the computed lines change after updating the price data. This may happen if the new price data contains some turnarounds that lead to a displacement of previously established lines. Hence, this may result in new sentiment values for the past so that the signal series may change.

In the case that a once optimized study should be used without re-optimization for a longer period this may lead to unpleasant results. Hence, you should not use the support/resistance sentimentors in this type of study.

19.3.3 The Parameters of the Support/Resistance-Sentimentor

Parameter:

Smoothness Span: speed of the MA of the MasterChart

Delta %: maximal distance of a turnaround's high/low from a price level

min. Tests: number of required tests of a price level

Interpretation:

smoothed price inside the lower half of the support region => sentiment value = 75

smoothed price inside the upper half of the support region => sentiment value = 65

smoothed price inside the lower half of the resistance region => sentiment value = 35

smoothed price inside the upper half of the resistance region => sentiment value = 25

19.4 Trendline Sentimentor

A trendline drawn into the MasterChart can work as a sentimentor. In addition to crossing a trendline the position of the current stock price relative to the trendline is also taken into account when computing the sentiment of a Trendline-Sentimentor.

The *support zone* is the zone between the trendline itself and the line resulting when displacing the trendline by *Support -Delta%*. Analogously, the *resistance zone* is defined as the zone between the trendline and the trendline displaced by *Resistance Delta %*.

In case the price is above the support zone or below the resistance zone, the sentimentor can be configured to expel a specific sentiment.

The following graphic shows the zones of a trendline:





Parameter:

Support-Delta %: for defining the support zone (0 = no support zone)

Resist-Delta %: for defining the resist zone (0 = no resist zone)

Always-Support: 1 = activation of Always-Support-Sentiment; 0 = deactivation

Always-Resist: 1 = activation of Always-Resist-Sentiment; 0 = deactivation

Interpretation:

Trendline is crossed from below => Sentiment = 100

Trendline is crossed from above => Sentiment = 0

Price inside the Support Zone => Sentiment = 75

Price inside the Resist Zone => Sentiment = 25

Price above the Support Zone and parameter "Always Support" is set to 1 => Sentiment = 65

Price below Resist Zone and parameter "Always Resist" is set to 1 => Sentiment = 35

Else: Sentiment = 50

19.5 Manually defined Sentimentors

A manually defined sentimentor provides the parameters *buy threshold* and *sell threshold* which are used for interpretation.

20 Cascading Studies

20.1 General Notes

Cascading Studies is a powerful new NanoTrader module that brings true intermarket analysis combined with multiple time frame analysis to the trader's desk. With the click of a mouse embed an existing study for, say, the Dax-Future, into a study for, say, the Eurostox-Future. Now the outcome of the Eurostox analysis is taken into account for trading the Dax.

NanoTrader allows unlimited flexibility when combining studies:

- Embed as many studies as you want into one (container)-study.
- The embedded studies can use any aggregation. So for example a longer-term trend study can be used as a filter for a short term swing strategy. Even non time-based aggregations like Renko or Point & Figure can be applied.
- An embedded study may contain embedded studies itself which allows for a complete hierarchy of *cascading* embedded studies.
- The complete range of backtesting and optimization tools is available for cascading studies. So taking the example from above, an optimization of the Dax study can also optimize the embedded Eurostox study.
- An embedded study behaves just like a "normal" sentimentor, so it can also be used as a Stop or as a Filter.

Among the possible applications of the *Cascading Studies* module are the following:

- Quite often a given symbol *A* follows in its direction another symbol *B*. So by trading A as soon as B emits a signal gives the trader the opportunity to trade as early as possible, even before A itself generates a technical signal.
- A study can embed other studies based on the same symbol but applying a different time frame. This *Multiple Time Frame* analysis allows to trade if there is the same signal in, say, the daily-based study, the hourly-based study, and the 1-minute study.
- Compute your own external sentiments and embed them into a NanoTrader study using DDE.
- Embed a number of studies for shares for trading a corresponding index future.

The following screenshot shows an example where the outcome of a FDAX study is used for trading the FESX:





Besides embedding a study NanoTrader also allows to embed price charts of other symbols into a container study. The embedded price charts follow the MasterChart in its aggregation and zoom. This allows for extremely convenient exploration of how different symbols behaved at a given time:





20.2 Using Cascading Studies

20.2.1 Embedding a Study

Parabolic SAR PFE Pivot Points Point & Figure ReentryFilter Renko RSI RSI-smoothed Slow Stochastic Stochastic SuperTrend Support/Resistance Study Time Price Opportunities TwinChart Volume	▲ >>	Crossing MA Volume Viewer Exp. Moving Average Time Price Opportunities Key Price Levels Volume Profile LiveStatistics Express\PrevDayClose Bollinger Bands Kaufmann's AMA Meta Sentimentor Express\EMA Trailing Stop Trailing Stop EoP MACD
Display in MasterChart	▼.	MACD

To embed a study, simply choose the "Study"-sentimentor from the Add Sentimentor dialog:

After adding the Study sentimentor a selection dialog will automatically show up:



Select Symbol	×
Type to search	ρ
 Commodities Equities ETFs FX Indices Indices Asia Indices Australia Indices Europe If EU Stocks 50 CFD If EU Stocks 50 Sep 21 CFD If France 40 CFD If France 40 Jul 21 CFD If Germany 30 CFD If Germany 30 CFD If Germany 30 CFD I 0_Min_Plain-20-Days 60_Min_Plain-20-Days 60_Min_Plain-20-Days ALIAS_TEST Bollinger 	•
OK Cancel	.:

Just highlight the study to be embedded and click OK.

To change the study assigned to a Study sentimentor doubleclick on it in the DesignerBar and the selection dialog will show up again.

The table of the DesignerBar will be displayed in yellow if it is embedded into another study. This is to make the trader aware that changing the settings of this study might lead to a new signal in the container study – which might be directly send to the exchange.

20.2.2 Embedding a Price Chart

Instead of embedding a complete study it is also very interesting to just embed a price chart of another symbol. An embedded price chart will follow the MasterChart in its aggregation and hence allows to view instantly how different symbols reacted at a certain time.

For embedding a price chart choose the "Study"-sentimentor from the Add Sentimentor dialog:



Type to search Parabolic SARPFEPivot PointsPoint & FigureRentryFilterRentryFilterRSIRSISISI	ρ ▲	× ↑ Crossing MA Volume Viewer Exp. Moving Average Time Price Opportunities Key Price Levels Volume Profile LiveStatistics Express\PrevDayClose Bollinger Bands Kaufmann's AMA Meta Sentimentor Express\EMA Trailing Stop Trailing Stop EoP MACD MACD	4
Oisplay in MasterChart Display as subwindow in MasterChart w nsert as: Sentimentor Filter	vindow	Tactic Close	Ţ

Then highlight the symbol to be embedded, e.g.:

Select Symbol ×
Type to search
Commodities
▷ Equities
▷ ETFs
⊳ FX
 Indices
Indices Asia
Indices Australia
Indices Europe
Indices UK
 Indices US
II US Dollar Index (per 0.01) Sep 21
II US Small Cap 2000 CFD
II US Small Cap 2000 Sep 21 CFD
▶ 🔢 US SP 500 CFD
▷ 🔢 US SP 500 Sep 21 CFD
II US Tech 100 CFD
II US Tech 100 Mini (Per 10) CFD
US Tech 100 Sep 21 CFD
Volatility Index Aug 21 CFD
Volatility Index Jul 21 CFD
OK Cancel

If the MasterChart shows a MarketProfile the embedded charts will also show a MarketProfile following the settings of the MasterChart.





20.2.3 Displaying an Embedded Price Chart in the MasterChart

A price chart can be overlayed in the MasterChart by selecting "Display Study in MasterChart" from its legend context menu.

[10 Min.] Study 'US SP 500 CF	-n'	(1_0)	
	[1]	Aggregation of Study 'US SP 500 CFD'	- F
	1	Study 'US SP 500 CFD' Visualization Settings \ldots	
	Ŧ	Display Study 'US SP 500 CFD' in MasterChart	
	×	Remove Study 'US SP 500 CFD'	43
	۶	Configure Study 'US SP 500 CFD' in DesignerBar	-

A price chart overlayed in the MasterChart will be displayed as if it was the MasterChart itself, i.e., there is no specific matching of the price scales. Any number of price charts may be displayed in the MasterChart:



Note that the InfoBar on the Eval page allows to see the prices of the embedded price charts below the mouse position at a glance.

20.3 Parameters of the Study-Sentimentor

The parameters of the Study sentimentor are ignored if only the price chart is displayed.

As opposed to all other parameters of the available sentimentors the parameters of the Study sentimentor are *not* altered within an optimization. Hence it is not necessary to define them as *constant* prior to an optimization.

When embedding a real study the parameters have the following meaning:

20.4 Parameter "Sentiment Mapping"

This parameter defines how to map the result of the embedded study into the container study. Two mapping styles are supported, where a value of "0" means "map signals" and a value of "1" means "as is".

20.4.1 Mapping Style "As Is"

The MetaSentimentor is taken without any change and becomes the sentiment of the Study sentimentor.

20.4.2 Mapping Style "Map Thresholds"

Recall that for generating a signal in a study the MetaSentiment of a given period must trigger a threshold (Enter Long, Enter Short, etc.) as defined in the "Trading"-line of the Designer table. Moreover, Filters may reject a potential signal. As the thresholds in the container study might be very different from those in the embedded study, just mapping the MetaSentimentor "as is" might not be sufficient to carry over signals into the container.

The mapping style "Map Thresholds" takes that into consideration by using the following mapping policy:

For a given period map the sentiment as follows:

- BLOCK sentiment in embedded study
 => BLOCK sentiment in container study
- FLAT sentiment in embedded study
 => FLAT sentiment in container study
- Long sentiment in embedded study and potential filters in embedded study allow a long position:
 => sentiment 100 in container study
- Short sentiment in embedded study and potential filters in embedded study allow a short position
 => sentiment 0 in container study



- Close Long sentiment in embedded study
 => sentiment 40 in container study
- Close Short sentiment in embedded study
 => sentiment 60 in container study
- Otherwise => sentiment 50 in container study

20.4.3 Using an Embedded Study as a Filter

A frequent application of Cascading Studies is to use an embedded study as a Filter, e.g, only go long based on a 5-minute FESX study if the embedded 60-minute FDAX trend study *allows* to go long.

Given the quite common scenario that the 60 minute FDAX study works with a couple of filters then it would be most comfortable if the filter settings could be embedded *directly* into the container – without having the need to also generate signals.

To achieve this, add the 60-minute study as a Filter to the 5-minute study and choose the mapping style to be "0", i.e., "Map Signals".

In this specific filter setup, NanoTrader applies the following mapping rule:

- Embedded Study in state BLOCK (light blue area in the MasterChart) => container study in state BLOCK (i.e., discard all signals)
- Embedded Study in state FLAT (dark blue area in the MasterChart) => container study in state FLAT (i.e., close position and discard all signals)
- Embedded Study allows Long (green area in the MasterChart) => sentiment = 100 in container study.
- Embedded Study allows Short (red area in the MasterChart) => sentiment = 0 in container study.
- Otherwise => sentiment 50 in container study

The following screenshot shows an example where a study using a trend filter is applied on a 60 minute FDAX study. This study is embedded in a 5-min FESX study:



NanoTrader - FESX MAR09 XEurex [5 Min.] TradeGuard	
Elle View Extras Portal Help	
 Copen a Chart Save All PageManager Tile Trade/Wizard SpeedTrader Time & Sales About 	
FDAX MAR09 XEurex [60 Min.] Trend FESX MAR09 XEurex [5 Min.] TradeGuard x	▼
Image: Section of the section of th	EDAX MAR09 XEurext (50 Min.) Trend Image: Second
Ready	

The visual indication from the 60-minute study might assist when trading manually or it filters signals when enhancing the 1-minute study with further sentimentors for automatic trading.

20.5 Parameter "Optimize Embedded Study"

If this parameter is set to 1 then the embedded study is also optimized whenever the container study is optimized.

20.6 Some Technical Details

20.7 Timescale Matching

As the timescales of the container study and embedded study will in general not be equal due to different aggregations it is important to understand how the periods of an embedded study are mapped onto the container study's time scale.

The general principle is as follows: Each period has a *begin-of-period timestamp* as well as an *end-of-period time stamp*. The sentiment generated by a given completed period p can be used in a container study starting from the earliest period having a begin-of-period timestamp equal to or later than the end-of-period timestamp of period p.

The following example demonstrates this. A 60-minute study is embedded in a 10-minute study:





At 19:59:59 the periods C and E are completed. The sentiment published by period E allows short signals –hence the red coloring. That sentiment carries over into the container study for period C and subsequent periods.

20.8 Inherent Difficulty with Timestamps

When matching timescales of different symbols onto each other there is unfortunately an inherent difficulty associated with timestamps of arriving ticks. Some data providers, such as eSignal, provide the original timestamp from the exchange with each incoming tick, while others, such as Patsystems, use the arrival time at the user's PC as a timestamp. Now suppose a study with a time aggregation of, say, 5 minutes which is embedded into a container study. As soon as the final period of the embedded study is completed the sentiment provided by that period is published to the container study. The difficulty comes with the question: "When is the period finished?"

There are essentially two ways to decide that question:

- 1. Wait until a tick with a time stamp arrives having a timestamp beyond the required end-of-period, i.e., a new period is created by that tick.
- 2. Check the time of the PC against the required end-of-period time.

Alternative 1 has the problem that it might take some seconds after the real end of the period until a new tick arrives. Hence the publishing of the sentiment has a delay and will overwrite the sentiment of the container study that was used for that period until the new sentiment is computed. Note that this might lead to a change in the signal in the container study.

Alternative 2 works perfect whenever the live feed uses the local PC's timestamps, as Patsystems does. A sentiment is published immediately into the container after the PC's clock signals the end of period.

Note, however, that when the live feed uses the exchange provided time stamps, ticks might arrive with a delay at the user's PC and thus might fall into a period that was already assumed to be completed – and hence the sentiment of that period was already published. Now, in a pathological case, the "delayed" incoming tick, which will be put into that already completed period, might

change the sentiment of that period and hence also a potential signal in the container study that was based on the original sentiment.

Overall, as Alternative 2 works perfect in many scenarios and has drawbacks only in rare cases with particular live feeds, this alternative is implemented by NanoTrader.

21 Walkforward Analysis with the SystemTester

21.1 Overview

The NanoTrader SystemTester is targeted at those traders that follow *trading systems*. The main problem with classical trading systems is their rigidity, i.e., they are not capable of "adjusting themselves" to changing market conditions. A once created trading system is kept unchanged until it becomes unprofitable. To tackle this problem, a process is required that adapts a system in a controlled way. This adaptation process itself then becomes an integral part of the trading system.

The NanoTrader-SystemTester allows to simulate this whole adaptation process. One starts with an initial study for an optimization period in the past. The SystemTester now optimizes the initial study for the chosen period. After this optimization, the computed sentimentor settings are applied on the subsequent *x* periods following the optimization period. Note that these *x* periods were not part of the optimization, hence this data is so-called "out of sample data". The resulting signals within these *x* periods are executed by the SystemTester and the corresponding trades are recorded. Now the optimization period is relocated for *x* periods in direction of the present and the steps optimization & signal execution are performed again.

At the end of this "walk forward" simulation, the resulting trades can be analyzed in terms of the equity curve, performance report, and trading report. The optimized sentimentor settings of the underlying study for the respective optimization periods are stored in a database and can also be analyzed. Thus, the complex process "Usage of a trading system with controlled adaptation" becomes completely transparent and can be optimized itself.

21.2 Starting the SystemTester

The SystemTester is started through the \mathfrak{S} icon of the DesignerBar.

System Test					×
Name: Simple	eTrading				
System based parameter do	d on the C mains.	Cross	ingMA with	n tight	*
Creation of S	tudies —				
Evaluator:	Future Tra	iding			
Initial study:				Т	ries:
25.09.2018	÷ ‡ 31.	.12.2	019 - ‡	0	
Walk-forward	end: 21	.06.2	021 - ‡	S	tart fixed
New optim	ization ev	ery	1	perio	d(s)
	Tri	ies:	25	Cre	ate
Computed St	udies				
From		То			
25.09.18 Tue 00 31.12.19 Tue 0 25.09.18 Tue 00 02.01.20 Thu 0					
16.11.18 F	Fri 00:00	25.	02.20 Tue	0	•
Simulation Follow signals	not older rage Ratir Simulate	than ng:E	2 517.92 Report	peri	iod(s).
Total net pr Total # of tr Winning trad Losing trade Percent pro Profit factor	ofit: 4 rades: 2 des: 1 es: 9 fitable: 6 : 1	464.9 23 14 3 50.87 1.07	90 7%		•
Save	Sa	ve &	Close		Close

The SystemTest-dialog is divided into four parts that are now discussed successively.

The first two fields are used to specify the name of the SystemTest and to enter notes.

System Test	×
Name: SimpleTrading	
System based on the CrossingMA with tight parameter domains.	
	T

NanoTrader shows entries for all created SystemTests for a security using their respective names in the WorkspaceBar. The icon used for SystemTests is s.





WorkspaceBar	,	×
⊳	ETFs	*
⊳	FX	
	Indices	
	Indices Asia	
	Indices Australia	
	 Indices Europe 	
	EU Stocks 50 CFD	
	EU Stocks 50 Sep 21 CFD	
	France 40 CFD	
	France 40 Jul 21 CFD	
	🔺 🔢 Germany 30 CFD	
	Plain 10_Min_Plain	
	🋃 1_Day_Plain	
	Pa Bollinger	
	🋃 JWSignalMagic	
	🛃 LoopTest	
	Plain	
	SimpleTrading	
	🖻 TradeGuard-10min	
	P TradeGuard	
	🛃 trend	
	📐 🛃 1_Min_Tradeguard	
	SimpleTrading	
	🖻 🖪 Germany 30 Mini (Per10	
	If Germany 30 Sep 21 CFD	
	If Germany 30 Tages Mon	
	Italy 40 CFD	Ŧ

To load a SystemTest simply doubleclick the corresponding entry in the WorkspaceBar.

21.3 Creation of Studies

The way signals of the underlying study have to be executed and rated are determined by an Evaluator. In the given example, the "Performance Trading"-Evaluator is chosen.

Creation of	Studies		
Evaluator:	Future Trading	-	

The Evaluator chosen in the SystemTester may differ from the Evaluator used in the underlying study, e.g., it is completely reasonable to use the Trendsignal-Evaluator in the study and to let the SystemTester trade the signals with the "Performance Trading"-Evaluator. In fact, this particular combination corresponds most closely with the overall NanoTrader-philosophy.

By clicking the -button the Evaluator-Settings-dialog is started.

The fields just below Initial Study are used to specify the initial optimization period as well as the number of tries used for optimizing the study on that period.

Initial study:							Tries:
25.09.2018	*	*	31.12.2019	*	*	0	

If the SystemTest-dialog is started through the menu of the DesignerBar then the currently selected optimization period of the DesignerBar is selected automatically. The usual approach to start the SystemTester is to load the MasterChart and study in question, select the optimization period with the mouse followed by an intensive optimization. If one is satisfied with the result the SystemTest-dialog is started and the initial optimization period is set automatically. The current sentimentor settings of the DesignerBar are used as the initial study by the SystemTester. As the sentimentors have just been optimized there is no need to perform another optimization by the SystemTester. Hence, the default setting of 0 Tries for optimizing the initial study is wanted the optimization effort in terms of Tries has to be specified.

The following settings define how the SystemTester adapts the optimization period after each "optimization/signal execution" iteration:

Walk-forward end:	21.06.2021 -		+ ÷	Start fixed
New optimization	n every 10			period(s)
	Tries:	25		Create

The Walk-forward end defines the date until which the end of the optimization period has to been moved successively.

If the beginning of the optimization period is to be kept unchanged within the whole process, the field Start fixed needs to be checkmarked. This leads to a successive enlargement of the optimization period with each iteration of the walk-forward process. If the beginning of the optimization period is kept unchanged the trades of the underlying study are rarely changed and this finally leads to a reduced number of signal changes in the SystemTester.

The setting New optimization every 1 period(s) defines that the end of the optimization period is moved forward one period after each iteration. When specifying 5, say, the result of an optimized study would be applied for the subsequent 5 periods *without* an optimization.

The Tries field specifies the optimization effort in terms of tries for each iteration.

Clicking the <u>Create</u>-button starts the creation of the studies. Depending on the settings, this process may take quite a while.

21.4 Visualization within the Study Creation Process

The creation of the studies can be examined in the SystemTest-Monitor. The process can also be aborted by clicking the Stop-button:



System Test			×
Computing from 07.01.20 until 12.04.21 improves: 0 lastImprove: 0 12.04.21 Mon 00:00: currently long	quote = 15244.0	account = 1310.60	•
Computing from 08.01.20 until 13.04.21 improves: 0 lastImprove: 0 13.04.21 Tue 00:00: currently long	quote = 15257.4	account = 1324.00	
Computing from 09.01.20 until 14.04.21 improves: 0 lastImprove: 0 14.04.21 Wed 00:00: currently long	quote = 15185.5	account = 1252.10	
Computing from 10.01.20 until 15.04.21 improves: 0 lastImprove: 0 15.04.21 Thu 00:00: currently long	quote = 15278.6	account = 1345.20	
•			▶
Γ	Stop		
	-		:

In case the optimization is performed at the end of each period (New optimization every 1 period(s)) the actual optimization period, the current position, and the equity at the end of the period are monitored. The X-Ray is used to show the current end of the optimization period in the charts. The MetaSentimentor, the signals and the charts are drawn with respect to the best found setting for the underlying study. In contrast, the Equity-window displays the equity that results from the trades simulated by the SystemTester.

In case the New optimization is not performed at the end of each period, the SystemTester first creates all studies without updating the charts and without displaying the current position. Only the X-Ray is used to show the end of the current optimization period.

21.5 Viewing the Created Studies

If the SystemTester finds an improved parameter setting for the sentimentors of the underlying study, this new setting is stored in the study database:

Computed Studies		
From	То	
13.05.19 Mon. 0	13.08.20 Thu 0	
19.06.19 Wed 0	21.09.20 Mon 0	
20.12.19 Fri 00:00	26.03.21 Fri 00:	-

The From/To-columns denote the optimization period for which an improved study has been computed. In the given example, the first optimization period ended at 08/13/20. The study found for this period has not been improved for the subsequent six weeks.

If a row in the table is clicked with the mouse, the corresponding study becomes *active* and all charts and the signals are drawn with respect to this study. Moreover, the parameters of the study are displayed in the DesignerBar and the Eval-page of the InfoBar displays the performance report for this study. You may also use the Arrow-Up/Arrow-Down keys on your keyboard to walk through the study database. This allows to visualize all studies step by step and to analyze the changes from study to study.

21.6 Executing the Signals of the Underlying Study

As soon as the study creation process has finished the simulated trades of the SystemTester are displayed in the MasterChart.

Besides the Evaluator, the execution of signals by the SystemTester is controlled by the following settings:

Simulation
Follow signals not older than 2 🌐 period(s).
Average Rating: 6519.57
Simulate Report

The field Follow signals not older than defines how old the last signal of the underlying study may be at maximum in order to be taken into account by the SystemTester. In case the last signal of the underlying study is older it will be ignored by the SystemTester, otherwise it will be converted into a corresponding opening or closing of a position.

Clicking the Simulate -button results in an application of the current Simulationsetting. This should be used to change the displayed signals to the SystemTester trades after examining a study from the study database.

To obtain a detailed trade report of the trades simulated by the SystemTester click the Report -button.

21.7 Performance Report

The lower part of the SystemTest-dialog displays the performance report of the system test:

Total net profit: Total # of trades: Winning trades: Losing trades: Percent profitable:	464.90 23 14 9 60.87%	A	
Profit factor:	1.07	*	

The Average Rating is the average rating of the underlying study computed over all optimization periods. The list below show the performance report of the trades executed by the SystemTester.

21.8 Saving and Closing the SystemTest-dialog

The current settings of the SystemTester including the study database can be saved. The SystemTest-files are automatically displayed in the WorkspaceBar



below their corresponding securities. If a SystemTest is stored for the first time, the WorkspaceBar has to be refreshed (F5) to show the new entry.

Save Save & Close Close	Save	Save & Close	Close
-------------------------	------	--------------	-------

21.9 Some Hints for Using the SystemTester

The creation of mechanical trading systems is a very time consuming process and requires that the trader is fully clear concerning the chosen settings and their consequences, e.g., besides the selection of the sentimentors and the restriction of their parameters the signal execution policy as specified in the Evaluator has an enormous influence of the trading result. Likewise, the way the optimization period is moved over the time scale is of great importance.

Future releases of NanoTrader will add more functionality for creating mechanical trading systems: filter, diverse stop policies, possibility for pyramiding, etc.

The majority of traders do not use mechanical trading systems, but follows the *discretionary* approach where an analysis and the resulting trades are situational. For these traders NanoTrader plays the role of an decision support system providing ideas and proposals for trades. Even for this kind of usage the SystemTester is of great advantage, because the "trade proposals" resulting from a given study and trading approach can be simulated over a long time period and can be adjusted to suit the individual needs of each trader.

22 Feedback

Fipertec is greatly interested in receiving feedback from you and to add functionality aiding you in your daily trading processes. Please do not hesitate to submit your wishes to <u>info@fipertec.com</u>.

Thank you for reading this document.