

Developer KIT

ChartNet.nl
Online streaming charts, koersen en technische analyse

iBacktest 1.0c
2004 editie

INHOUD

iBacktest presentatie.....	2
iBacktest commando's.....	3
Order simulatie.....	3
Stop-loss programmering.....	7
Backtest strategieën.....	8
Status variabelen.....	10
Variabelen van opvolgende posities.....	12
De relatie tussen bars en de laatst uitgevoerde order.....	14
Voorbeelden.....	16
Penny stocks.....	16
Geoptimaliseerde instap.....	18
Trend detectie op koers.....	20
Sell in may and go away !.....	21
Intraday uitbraak.....	22

iBacktest presentatie

iBacktest is de backtesting module van ChartNet. Het is een uitbreiding op de IBuilder taal. Het stelt u in staat om trading strategieën te maken en te testen gebruikmakend van de bekende en de met iBuilder zelf geprogrammeerde indicatoren.

I Bcktest lijkt veel op de programmeertaal BASIC. U kunt eigen systemen maken gebruik makend van de koersen die door ChartNet worden verstrekt:

- 1• Openenings koers van elke bar/candle
- 2• Slot koers van elke bar/candle
- 3• Hoogste koers van elke bar/candle
- 4• Laagste koers van elke bar/candle
- 5• Volume van elke bar/candle

Bars/candles zoals te zien in de ChartNet Software.

Een pijl geeft elke door het system gegenereerde koop- of verkoop order aan. Deze pijlen geven de aan- of verkoop prijzen aan van de gesimuleerde orders. Bovendien berekend IBacktest een curve die de performance van het systeem laat zien. Deze curve wordt als een indicator boven de koersgrafiek afgebeeld.

IBacktest commando's

Order simulatie

Een order is gedefinieerd als: de karakteristiek van de de transactie (koop/verkoop), het betreffende aantal aandelen, de condities van de uitvoering en de datum van de uitvoering. Een of meer parameters mag ontbreken.

De karakteristiek van de de transactie

Buy, sell

BUY *count* **SHARES** (**AT MARKET** | **AT price LIMIT** | **AT price STOP**)

Dit commando opent een long positie. Als de backtest portefeuille short zit wanneer een dergelijke order wordt uitgevoerd, dan wordt de short positie ook gekocht om een long positie van *count* aandelen te krijgen. Als de backtest portefeuille al long zit als een dergelijke order wordt uitgevoerd, dan hangt het af van de ingevoerde money management regels.

SELL [*count* **SHARES**] (**AT MARKET** | **AT price LIMIT** | **AT price STOP**)

Dit commando sluit een long positie. Als de backtest portefeuille leeg is wanneer een dergelijke order wordt uitgevoerd, dan gebeurt er niets. Als de backtest portefeuille long zit, dan worden *count* aandelen verkocht. Ook indien *count* groter is dan het aantal in bezit zijnde aandelen wordt alleen het aantal in bezit zijnde aandelen verkocht en wordt niet short gegaan. Als de *count* parameter ontbreekt dan worden alle aandelen verkocht.

Sellshort, exitshort

SELLSHORT *count* **SHARES** (**AT MARKET** | **AT price LIMIT** | **AT price STOP**)

Dit commando opent een short positie. Als de backtest portefeuille long zit wanneer een dergelijke order wordt uitgevoerd, dan wordt de short positie ook verkocht om een short positie van *count* aandelen te krijgen. Als de backtest portefeuille al short zit als een dergelijke order wordt uitgevoerd, dan hangt het af van de ingevoerde money management regels.

EXITSHORT [*count* **SHARES**] (**AT MARKET** | **AT price LIMIT** | **AT price STOP**)

Dit commando sluit een short positie. Als de backtest portefeuille leeg is wanneer een dergelijke order wordt uitgevoerd, dan gebeurt er niets. Als de backtest portefeuille short zit, dan worden *count* aandelen gekocht. Ook indien *count* groter is dan het aantal geshorte aandelen, wordt alleen het aantal geshorte aandelen gekocht en wordt niet long gegaan. Als de *count* parameter ontbreekt dan worden alle aandelen gekocht.

Aantal aandelen

Het aantal aandelen kan worden ingevoerd as een aantal cash eenheden als deel van het kapitaal of als onderdeel van de beschikbare cash.

- **SHARES** aantal aandelen
- **CASH** aantal in cash unit (zoals € of \$)
- **%CAPITAL** deel van het huidige kapitaal (zoals te zien in de equity curve)
- **%LIQUIDITY** deel van de huidige beschikbare cash

Voorbeeld Cash

REM Buy 1000€ (if quotes are expressed in €)

BUY 1000 Cash AT MARKET

Voorbeeld %Capital

REM Buy with 70% of the curret capital (shown by the equity curve for each bar)

BUY 70 %Capital AT MARKET

Voorbeeld %Liquidity

REM Buy with 40% of the current available cash

BUY 40 %Liquidity AT MARKET

Opmerking : Wanneer u de markt verlaat (*SELL*, *EXITSHORT*) is het aantal aandelen optioneel.

REM Exits all long positions

SELL AT MARKET

Uitvoerings conditie

Er zijn 3 verschillende order beschikbaar : market price, beste limiet en stop orders.

- **AT MARKET** market price
- **AT price LIMIT** beste limiet
- **AT price STOP** stop orders

Voorbeeld AT MARKET

REM Buy 100 shares at market price

BUY 100 Shares AT MARKET

Voorbeeld LIMIT

REM Buy 100 shares at the best limit 15.45€

BUY 100 Shares AT 15.45 LIMIT

Voorbeeld STOP

REM Buy 100 shares with a stop order at 16.18€

BUY 100 Shares AT 16.18 STOP

De datum van uitvoering

Indien het niet wordt aangegeven worden order gesimuleerd op de volgende bar. Echter, orders "at market" kunnen worden gesimuleerd voor of na die bar, dankzij een van de volgende keywords:

- **ThisBarOnClose** aan het slot van de huidige bar
- **NextBarOpen** aan het begin van de volgende bar (**standaard order**)
- **NextBarClose** aan het slot van de volgende bar
-
- **TodayOnClose** aan het slot van de de huidige dag (gebruikt in intraday)
- **TomorrowOpen** aan het begin van de volgende dag (gebruikt in intraday)
- **TomorrowClose** aan het slot van de volgende dag (gebruikt in intraday)

Voorbeeld ThisBarOnClose

REM Buy 100 shares at the close of the current bar
BUY 100 Shares AT MARKET ThisBarOnClose

Voorbeeld NextBarClose

REM Buy 100 shares at the close of the next bar
BUY 100 Shares AT MARKET NextBarClose

Voorbeeld TodayOnClose

REM Buy 100 shares at the close of the current day (intraday)
BUY 100 Shares AT MARKET TodayOnClose

Voorbeeld TomorrowOpen

REM Buy 100 shares at the open of the next day (intraday)
BUY 100 Shares AT MARKET TomorrowOpen

Voorbeeld TomorrowClose

REM Buy 100 shares at the close of the next day (intraday)
BUY 100 Shares AT MARKET TomorrowClose

Stop-loss programmering

Set Stop

SET STOP *price*

Dit commando stelt u in staat een custom stop die gebruik maakt van uw eigen algoritmes toe te voegen. Alle 4 de gebruikelijke stop order kunnen worden gesimuleerd in het iBacktest venster.

Voorbeeld Set Stop

```
REM A following stop
```

```
IF close > AVERAGE[30] AND NOT OnMarket THEN
```

```
  BUY 100 Shares AT MARKET
```

```
  REM Compute the difference between ideal price and real price
```

```
  Spread = openOfNextBar - low
```

```
  REM Initialize the highest quote since the entry on market
```

```
  Up = openOfNextBar
```

```
ENDIF
```

```
REM Updates the stop to let its distance to the highest price constant
```

```
IF OnMarket THEN
```

```
  Up = MAX(Up, high)
```

```
  SET STOP (Up - Spread)
```

```
ENDIF
```

Backtest strategieën

AS

Commando's en variabelen zijn verbonden aan een of meerdere strategieën. U kunt zo veel strategieën definiëren als u wilt. Indien u nergens naar een strategie verwijst in uw code, dan zal iBacktest 1 strategie creëren en worden alle commando's er naar verwezen. Indien u diverse strategieën binnen het zelfde systeem wilt definiëren dan dient u het keyword **AS** te gebruiken.

Voorbeeld orders verwijzen naar 1 strategie

```
REM Strategy named "moving average"  
  
IF close > AVERAGE[30](close) AND NOT OnMarket THEN  
    BUY 100 Shares AT MARKET AS « moving average»  
ENDIF
```

Elke positie karakteristiek wordt geëvalueerd inde context van zijn strategie. U kunt bijvoorbeeld een overall long positie in de markt hebben terwijl een u voor een bepaalde strategie short zit.

Voorbeeld orders verwijzen naar al de strategieën

```
REM Order computed on all the strategies  
SELL 50 Shares AT MARKET
```

De verkoop order (sell) wordt gesimuleerd in al uw strategieën. Indien u 2 strategieën heeft de verkoop op $2 * 50$ shares = **100** shares. Het is daarom zinvol om al uw long posities te verkopen met één instructie.

Deze principes zijn van toepassing op elke backtesting variabele of commando.

Voorbeeld EntryIndex, strategieën

```
REM Strategy "moving average"  
  
IF close > AVERAGE[30](close) AND NOT OnMarket THEN  
    BUY 100 Shares AT MARKET AS « moving average»  
ENDIF  
  
REM Sell on the breakout of the low of the entry bar
```

```
IF close < low[EntryIndex AS « moving average»] THEN  
    SELL AT MARKET AS "moving average"  
ENDIF
```

Status variabelen

OnMarket, LongOnMarket, ShortOnMarket

Deze variabelen stellen u in staat om de status van de backtest portfolio te bepalen. De status van de portfolio kan zijn: long, short of leeg.

Omschrijving

De status variabelen zijn belangrijk omdat ze de huidige status van de backtest portfolio laten zien.

In de meeste gevallen wanneer u een strategie backtest, koop- of verkooporders worden uitgevoerd afhankelijk van de status van de huidige positie in de markt. Bijvoorbeeld: hoewel er in beide situaties sprake is van verkoop is short gaan wat anders dan een exit van een long positie. In het eerste geval wordt een positie geopend en in het tweede geval wordt een positie gesloten.

De met de uitvoering van orders geassocieerde condities zijn niet altijd het zelfde. Bijvoorbeeld, u wilt een positie openen vanwege een technisch signaal en u sluit de positie vanwege money management overwegingen.

Voorbeeld OnMarket

```
REM Buy on moving average breakout
```

```
c1 = close > AVERAGE[30](close)
```

```
REM This condition is sufficient to enter the market
```

```
IF NOT OnMarket THEN
```

```
    IF c1 THEN
```

```
        BUY 10 SHARES AT MARKET
```

```
    ENDIF
```

```
ENDIF
```

```
REM But we have 2 conditions to exit
```

```
IF OnMarket THEN
```

```
    REM The following adds a condition given by a « new low » breakout
```

```
    C2 = close < LOWEST[10](low[1])
```

```
    REM The exit depends on a double condition
```

```
    IF NOT c1 OR c2 THEN
```

```
SELL 10 SHARES AT MARKET
```

```
ENDIF
```

```
ENDIF
```

Variabelen van opvolgende posities

CountOfLongShares, CountOfShortShares, CountOfPosition

Deze variabelen zijn:

- Het aantal aandelen van een long positie (0 indien niet long)
- Het aantal aandelen van een short positie (0 indien niet short)
- Het aantal van posities (als een piramide is toegestaan)

Omschrijving

Deze variabelen geven een preciezer informatie dan de status variabelen. Ze stellen u in staat uw huidige positie in de markt te weten, maar ze geven ook het aantal aandelen in de markt en het aantal uitgevoerde orders.

Voorbeeld CountOfLongShares, CountOfPosition

```
REM Buy on moving average breakout

c1 = close > AVERAGE[30](close)

REM This condition is sufficient to enter the market
IF NOT OnMarket THEN

    IF c1 THEN
        BUY 10 SHARES AT MARKET
    ENDIF

ENDIF

REM Pyramids 3 times if the buy condition is still true
IF OnMarket THEN

    REM The following adds a condition given by a « new low » breakout
    c2 = close < LOWEST[10](low[1])

    REM The exit depends on a double condition
    IF NOT c1 OR c2 THEN
        SELL CountOfLongShares SHARES AT MARKET

    REM Pyramids 3 times until (while the exit contion is false)
    ELSIF CountOfPosition < 3 THEN
        BUY 10 SHARES AT MARKET
```

ENDIF

ENDIF

Opmerking :

Piramides zijn alleen toegestaan als de optie « cumulate positions » is geactiveerd in de hoofd management sectie van het iBacktest venster.

Als u de optie « 1 stop voor alle posities » aanvinkt, dan worden alle posities samengevoegd tot een, en kan **CountOfPostion** niet groter zijn dan 1.

De relatie tussen bars en de laatst uitgevoerde order

EntryIndex

Dit is de index van de bar waarop de laatste order is uitgevoerd

Omschrijving

Deze constante stelt u in staat om de bar waarop het systeem een positie aanging vast te stellen en een daar aan gerelateerde stop te programmeren.

Voorbeeld EntryIndex

```
REM Buy in case of a moving average breakout
IF NOT OnMarket THEN
    IF close > AVERAGE[30](close) THEN
        BUY 100 %CAPITAL AT MARKET
    ENDIF
ENDIF

REM Exit under the lowest price of the candlestick of entry
IF OnMarket THEN
    SELL AT low[EntryIndex] STOP
ENDIF
```

EntryQuote

Dit is de uitvoer prijs van de laatst uitgevoerde order

Omschrijving

Dit is de variabele waarmee u een stop kunt programmeren gerelateerd aan het laatste instap punt.

Voorbeeld EntryQuote

```
REM Buy in case of a moving average breakout
IF NOT OnMarket THEN
    IF close > AVERAGE[30](close) THEN
        BUY 100 %CAPITAL AT MARKET
    ENDIF
ENDIF

REM Exit under the entry price
IF OnMarket THEN
```

```
SELL AT EntryQuote STOP
```

```
ENDIF
```

Voorbeelds

Penny stocks

Omschrijving

Hier is een voorbeeld van een long systeem dat orders genereert en interessante kansen detecteert.

Het is gebaseerd op een filter dat naar de absolute waarden van koersen kijkt. Dit is een "penny stocks" filter dat voorkomt dat er iets wordt gedaan met koersen hoger dan 10 (€, \$, £...)

De instap conditie is een nieuwe uitbraak omhoog, bevestigd door een opwaartse trend van de moving average. Het uitstap signaal is een neerwaartse doorbraak van een moving average (we gebruiken een moving average om het aantal valse signalen te verminderen).

Waarschuwing: Dit systeem hoeft niet winstgevend te zijn. We willen hiermee het programmeren van strategieën illustreren. U dient het te optimaliseren of mogelijk zelfs de aan- en verkoop condities volledig veranderen.

Voorbeeld PennyStocks

REM BUY CONDITION

REM if the highest of the bar higher than the highest of the 10 previous bars ?

c1 = high > highest[10](high[1])

REM is the highest of the bar higher than the 13bar exponential moving average of close ?

c2 = high > exponentialAverage[13](close)

REM is the close price lower than 10 ?

c3 = close < 10

REM is the candlestick white ?

c4 = close > open

IF c1 AND c2 AND c3 AND c4 THEN

BUY 70 %capital AT MARKET

ENDIF

REM EXIT CONDITION

REM does the low of the bar crosse under the 30bars exponential moving average of low ?

```
IF low CROSSES UNDER exponentialAverage[30](low) THEN
```

```
    SELL AT MARKET
```

```
ENDIF
```

Geoptimaliseerde instap

Omschrijving

Hier is een voorbeeld van een long systeem. Het is er op gericht alleen een long positie aan te gaan bij een voldoende daling binnen een uptrend.

De trend wordt gedetecteerd door een 26 bars gewogen moving average

Om op de « beste » koers in termen van rentabiliteit en risico, gebruiken we de « Parabolic SAR » indicator. Het is een trend volgende indicator.

Het bevestigt de uptrend en geeft ons het laagste koersniveau waarbij de trend niet omkeert.

We willen kopen op het niveau van de “Parabolic SAR”. Het risico is dat zo’n niveau wordt gebroken. Echter, als de conditie van het 26 bars gewogen moving average er bi wordt betrokken, dan hebben we goede mogelijkheden om een degelijke long trade te detecteren. Daarbij is het uiteraard ook per definitie zo dat een trend meer de neiging heeft voort te duren dan om te keren. We stappen uit als de gewogen moving average wordt gebroken.

Waarschuwing: Dit systeem hoeft niet winstgevend te zijn. We willen hiermee het programmeren van strategieën illustreren. U dient het te optimaliseren of mogelijk zelfs de aan- en verkoop condities volledig veranderen.

Voorbeeld Optimized entry

```
REM Parabolic SAR as trade following indicator
```

```
follow = SAR
```

```
REM is the close higher than the 26 bars weighted moving average ?
```

```
ha1 = close > weightedaverage[26](close)
```

```
REM is the SAR indicator under the close?
```

```
ha2 = close > follow
```

```
IF ha1 THEN
```

```
    IF ha2 THEN
```

```
        BUY 80 %capital AT follow LIMIT
```

```
    ENDIF
```

```
ELSE
```

```
    SELL AT MARKET
```

```
ENDIF
```


Trend detectie op koers

Omschrijving

Hier is een long/short systeem dat u in staat stelt om altijd in de markt te zijn. Het doel van het systeem is om de positie te draaien om te verdienen op koop- en verkooppaniek.

We trachten deze paniek te detecteren door naar de koers te kijken. We nemen aan dat er voorspellende waarde uitgaat van de toename van volatiliteit. Onze hypothese: abnormaal sterke variatie betekent paniek.

De money management is :70% van middelen voor long posities en 40% op short posities. Short posities zouden minder risico vol moeten zijn omdat het bekend is dat paniek in neerwaartse richting belangrijker is dan opwaarts. Money management is echter lastiger met short posities (potentieel verlies is ongelimiteerd !)

Waarschuwing: Dit systeem hoeft niet winstgevend te zijn. We willen hiermee het programmeren van strategieën illustreren. U dient het te optimaliseren of mogelijk zelfs de aan- en verkoop condities volledig veranderen.

Voorbeeld Trend detected on prices

```
REM compute the average variation of quotes
```

```
avt = AverageTrueRange[20](close)
```

```
REM buy condition: strong increase of prices (=panic buy)
```

```
IF close > close[1] + avt THEN
```

```
    BUY 70 %capital AT MARKET
```

```
ENDIF
```

```
REM short condition : strong decrease of prices (=panic sell off)
```

```
IF close < close[1] - avt THEN
```

```
    SELLSHORT 40 %capital AT MARKET
```

```
ENDIF
```

Sell in may and go away !

Omschrijving

Het is ook mogelijk om een system te bouwen, dat niet is gebaseerd op koersen, maar op statistieken.

Een bekende “beurswijsheid” is « sell in may and go away ! ». Een langjarige studie van de markten concludeert de slechtste performance in de maanden mei t/m september.

We willen het controleren met een trading system. Het principe is simpel: Kopen in oktober en short in mei.

Waarschuwing: Dit systeem hoeft niet winstgevend te zijn. We willen hiermee het programmeren van strategieën illustreren. U dient het te optimaliseren of mogelijk zelfs de aan- en verkoop condities volledig veranderen.

Voorbeeld Trend gedetecteerd op maanden (!!)

```
REM Sell short in may !
IF Month = 5 THEN

    SELLSHORT 50%capital AT MARKET

REM Buy in october
ELSIF Month = 10 THEN

    BUY 50%capital AT MARKET

ENDIF
```

Intraday uitbraak

Omschrijving

Laten we een beroemd intraday system beschouwen. Het is een uitbraak system gebaseerd op de limieten afgegeven door de 2 eerste bars van de dag,

Een instep in de markt wordt bepaald aan de hand van de uitbraak van een van deze limieten.

De uitstap vindt plaats om 16:00 uur. (locale tijd)

Waarschuwing: Dit system hoeft niet winstgevend te zijn. We willen hiermee het programmeren van strategieën illustreren. U dient het te optimaliseren of mogelijk zelfs de aan- en verkoop condities volledig veranderen.

Voorbeeld Intraday break out

```
REM At the close of the second bar of the day (index 1 because it starts at 0)
```

```
IF intradayBarIndex = 1 THEN
```

```
    REM Compute the levels of the highest and the lowest of the 2 first bars of the day
```

```
    up = Highest[2](high)
```

```
    down = Lowest[2](low)
```

```
ENDIF
```

```
REM Buy / Sell on breakout between the 3rd bar and 16 o'clock (local hour)
```

```
IF intradayBarIndex > 1 AND Time < 160000 THEN
```

```
    REM Breakout of the highest
```

```
    IF close > up THEN
```

```
        BUY 70%capital AT MARKET
```

```
        SELL AT MARKET TodayOnClose
```

```
    REM Breakout of the lowest
```

```
    ELSIF close < down THEN
```

```
        SELLSHORT 70%capital AT MARKET
```

```
        EXITSHORT AT MARKET TodayOnClose
```

```
    ENDIF
```

ENDIF