

## ADX

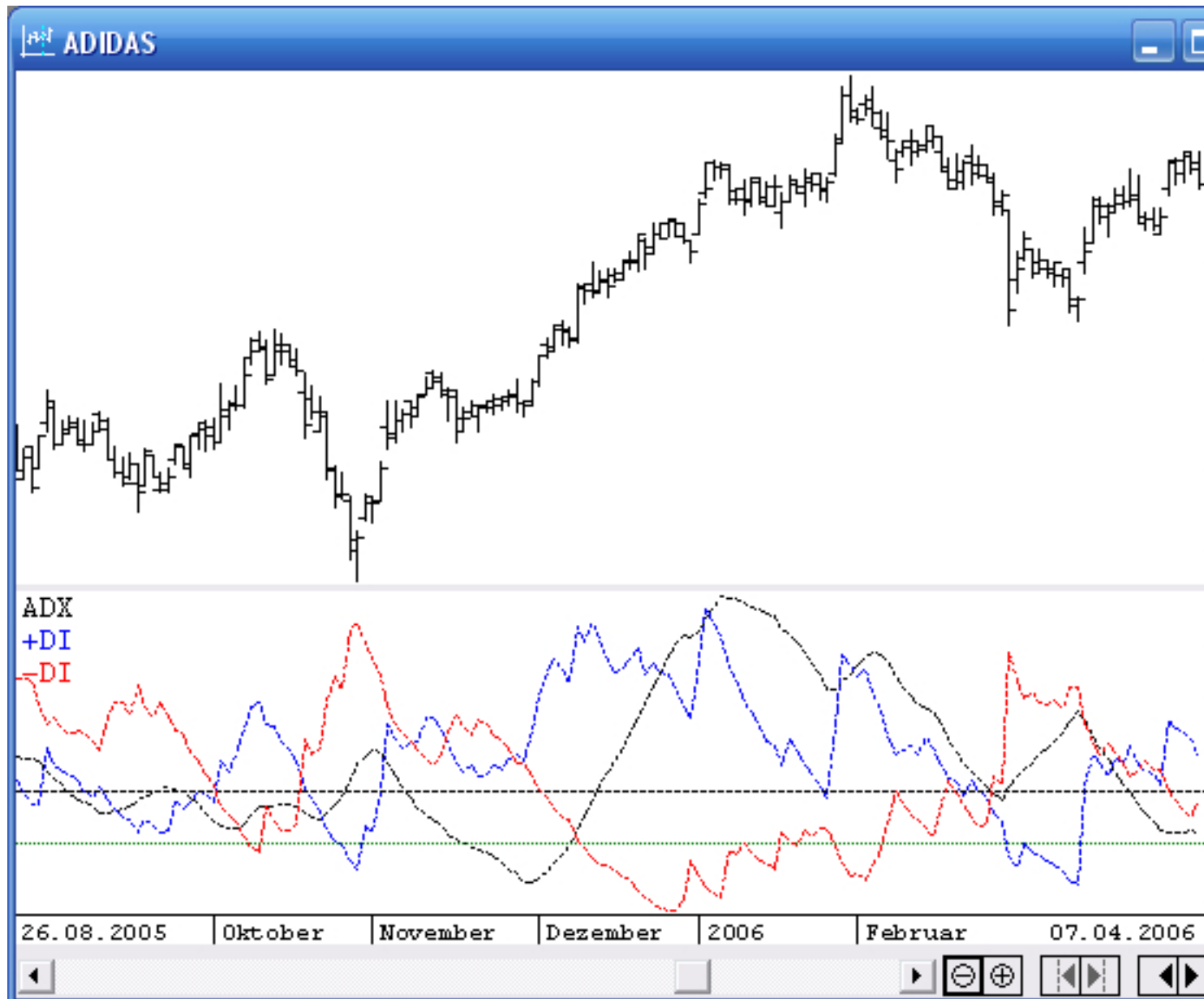
Geschrieben von: Reiner Reusch

Dienstag, 01. Dezember 2009 um 13:35 Uhr - Aktualisiert Dienstag, 01. Dezember 2009 um 13:54 Uhr

## Warnung

Ich übernehme keinerlei Haftung für das Funktionieren dieses Beispiels in der Praxis. Hier soll nur gezeigt werden, wie mit Eusdoni ein Handelssystem entwickelt werden kann.

## ADX-Beispiel



## Handelsansatz

## ADX

Geschrieben von: Reiner Reusch

Dienstag, 01. Dezember 2009 um 13:35 Uhr - Aktualisiert Dienstag, 01. Dezember 2009 um 13:54 Uhr

---

Wenn der ADX steigt und über einem bestimmten Niveau (zum Beispiel 15 oder 20) liegt, zeigt dies einen Trend an. +Di und -Di wird verwendet, um die Trendrichtung zu bestimmen.

## Umsetzung Wertpapiere

Es wird mit den im DAX enthaltenen Wertpapieren der Kursdatenbank getestet.

### Einstieg:

Es wird gekauft, wenn nicht schon gekauft wurde, +Di über -Di liegt und der ADX steigt.

```
function Einsteigen () : boolean;
begin
  if LongCount () = 0 AND LookForEnter AND
    ADX_Trend (ADX_Periods, ADX_TrendLevel) AND
    PDI (ADX_Periods) > MDI (ADX_Periods) then
    begin
      Einsteigen := true;
      LookForEnter := false;
    end
  else if ADX (ADX_Periods) < ADX_TrendLevel then
    begin
      { Der ADX muss erst wieder unter ADX_TrendLevel fallen,
        bevor nach einem neuen Einsteig gesucht wird. }
      LookForEnter := true;
    end;
  end;
```

Dabei ist **ADX\_Trend** folgendermassen definiert:

```
function ADX_Trend (Periods:integer; ADX_TrendRange:real) : signal;
begin
  ADX_Trend := RoC (ADX (Periods), 1) > 1.0 AND ADX (Periods) >= ADX_TrendRange;
end;
```

### Ausstieg:

```
function Aussteigen () : boolean;
begin
  { Aussteigen wenn der ADX unter MDI fällt oder
    MDI auf oder über das Niveau von PDI steigt. }
  if (RoC (ADX (ADX_Periods), 1) <= 0.0 AND ADX (ADX_Periods) < MDI (ADX_Periods)) OR
    MDI (ADX_Periods) >= PDI (ADX_Periods) then
    Aussteigen := true;
```

end; **Parameter einfach ändern.**

Für einfache Veränderung der Parameter werden Variable verwendet. Eine Funktion initialisiert die Variablen.

```
function VariableInit () : real;
```

```
begin
```

```
  ADX_Periods := 18;
```

```
  ADX_TrendLevel := 20.0;
```

```
end;
```

**Alle Wert im Überblick**

- ☐ 12 : Bedingungen und Werte
- ☉ Kaufen: Einsteigen () AND Date () > 19950701
  - ☉ Verkaufen: Aussteigen ()
  - ☉ Leer verkaufen: false
  - ☉ Leerverkauf glattstellen: false
  - 12 Anzahl Wertpapiere beim Einstieg: AnzahlWertpapiereBeimKauf ()
  - 12 Anzahl Wertpapiere beim Ausstieg: SecurityCount ()
  - 12 Einstiegskosten (Provision, ...): Max (10.0, EnterCount () \* EnterQuote () \* 0.0025)
  - 12 Ausstiegskosten (Provision, ...): Max (10.0, LeaveCount () \* LeaveQuote () \* 0.0025)
  - 12 Maximaler Einstiegskurs: OpenTomorrow ()
  - 12 Minimaler Einstiegskurs: NaN
  - 12 Maximaler Ausstiegskurs: NaN
  - 12 Minimaler Ausstiegskurs: OpenTomorrow ()
  - ☉ Ende des Tages: false
  - 12 Wertpapierertest Start: VariableInit ()
  - 12 Wertpapierertest Ende: NaN

- 0.25 % Provision. Mindestens aber 10.
- Kauf und Verkauf zum Eröffnungskurs des nächsten Tages.

## Resultate

### Übersicht

Gewinn: 61.711,74 Konto Max./Min: 68.823,27/-208.147,94 Gewinn Max./Min:

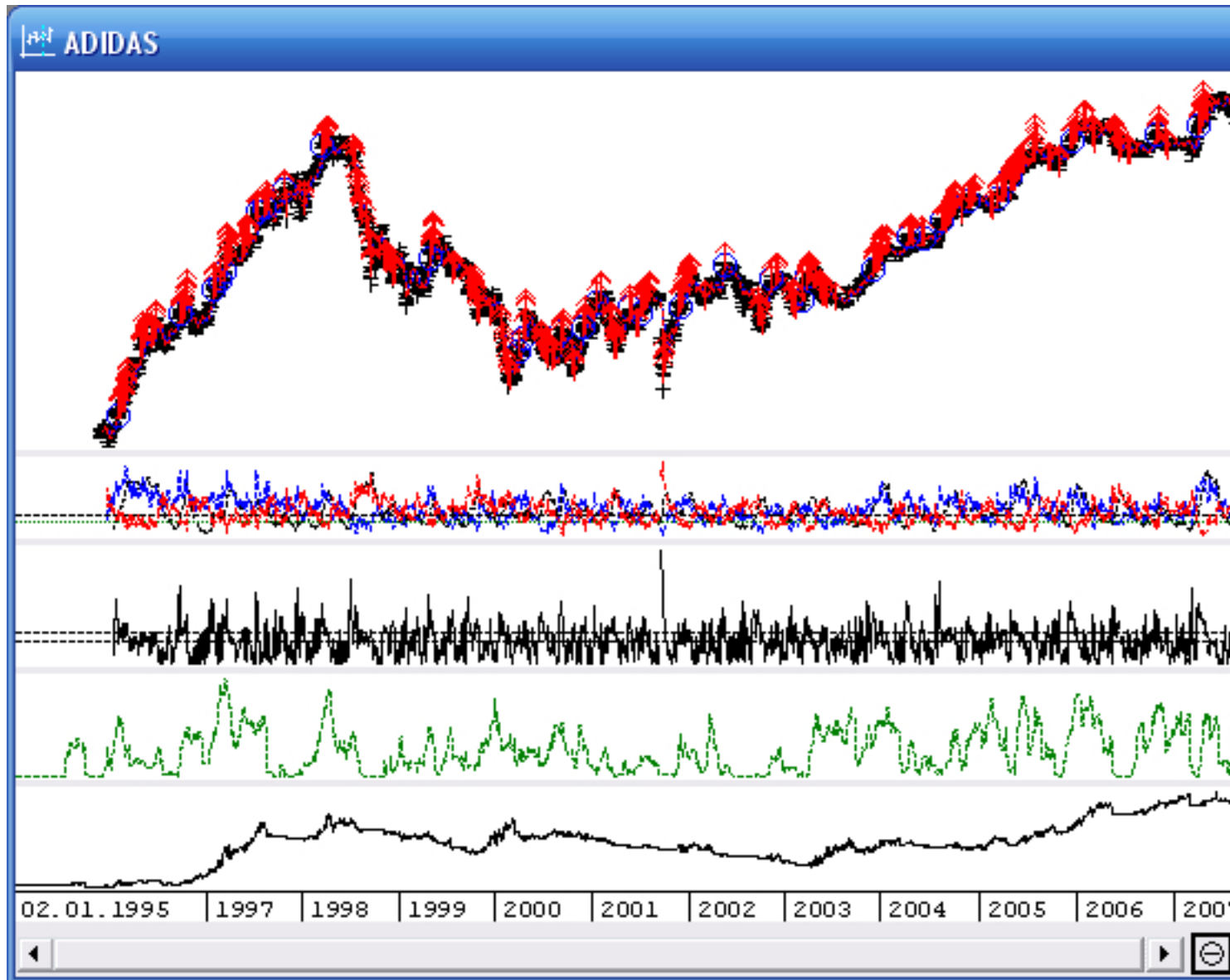
77.586,15/-2.836,35 Trans.: 1634 Gewinn/Trans.: 37,77

Ausstiege: 810 Gewinntrades: 34 % Durchschn. Gewinn: 10,01 % Durchschn. Verlust: 4,24 %

Das Ergebnis überzeugt nicht. Vor allem der minimale Kontostand von -204.147.94 Euro sagt

aus, dass man über 200.000 Euro benötigt hatte, wenn man alle Transaktionen genau nach den Vorgaben des Handelssystems ausgeführt hätte. Aber immerhin macht dieses einfache Handelssystem einen Gewinn.

### Chart



[Volle Bildgröße](#)

### Ideen für Verbesserungen

Im Handelssystem wird oft gekauft und verkauft. Durch den Einsatz zusätzlicher Indikatoren oder Signale, kann man eventuell Verlusttrades herauszufiltern.

# Download des kompletten Beispiels

[ADX\\_DAX\\_Long.tsy](#)