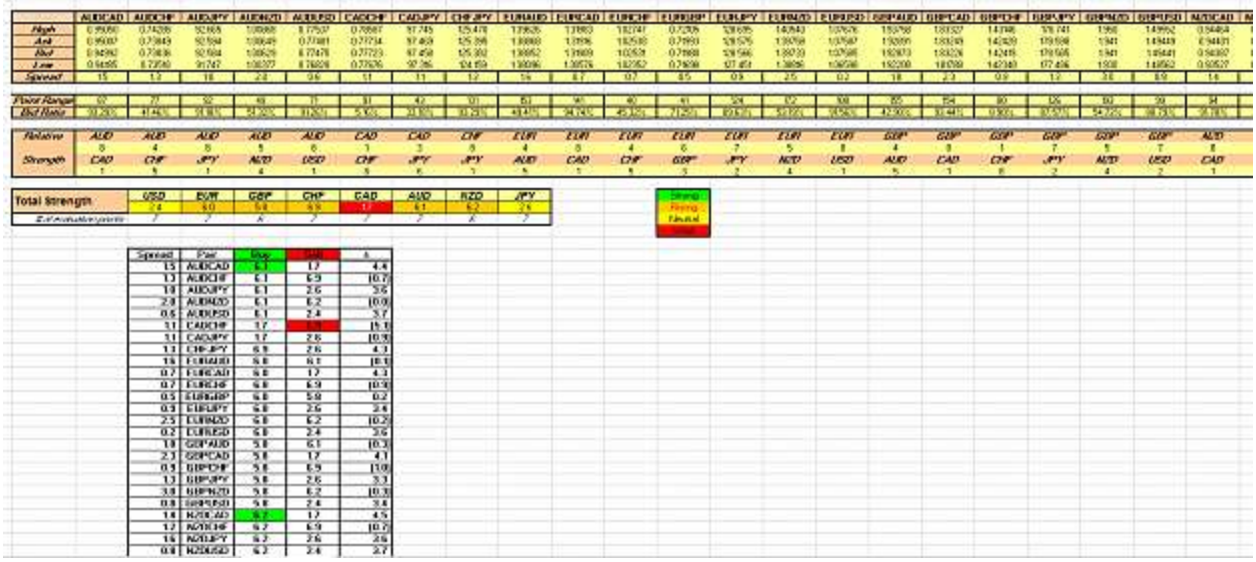
There is a quite well known Excel spreadsheet that uses the DDE capabilities of MT4 to calculate the currency strength using a formula based on the D1 high, D1 low and bid:



As a spin-off from a different project I'm working on, I have ported this to MT4.

It is a simple **indicator** that should replicate the results of the excel spreadsheet.
It has a few other enhancements (alerts, ability to select other timeframes for the calculation, click-to-open-with-template-applied for each pair) but overall it is a relatively simple thing.

I'm creating a thread here because:
1. A few different systems on FF use the currency strength meter so it makes more sense to have a single thread for it; and
2. It avoids going off-topic on those threadsIt is an element of the strength calculation, which I display for quick reference.

It is how far the current Bid price is to the high or low.

A green bar (high value) means that price is close to the high.

A red bar (low value) means that price is close to the low.

Think of it as a candle lying on its side.

Hope that helps

Bid ratio display modified

Why? It wasn't particularly intuitive

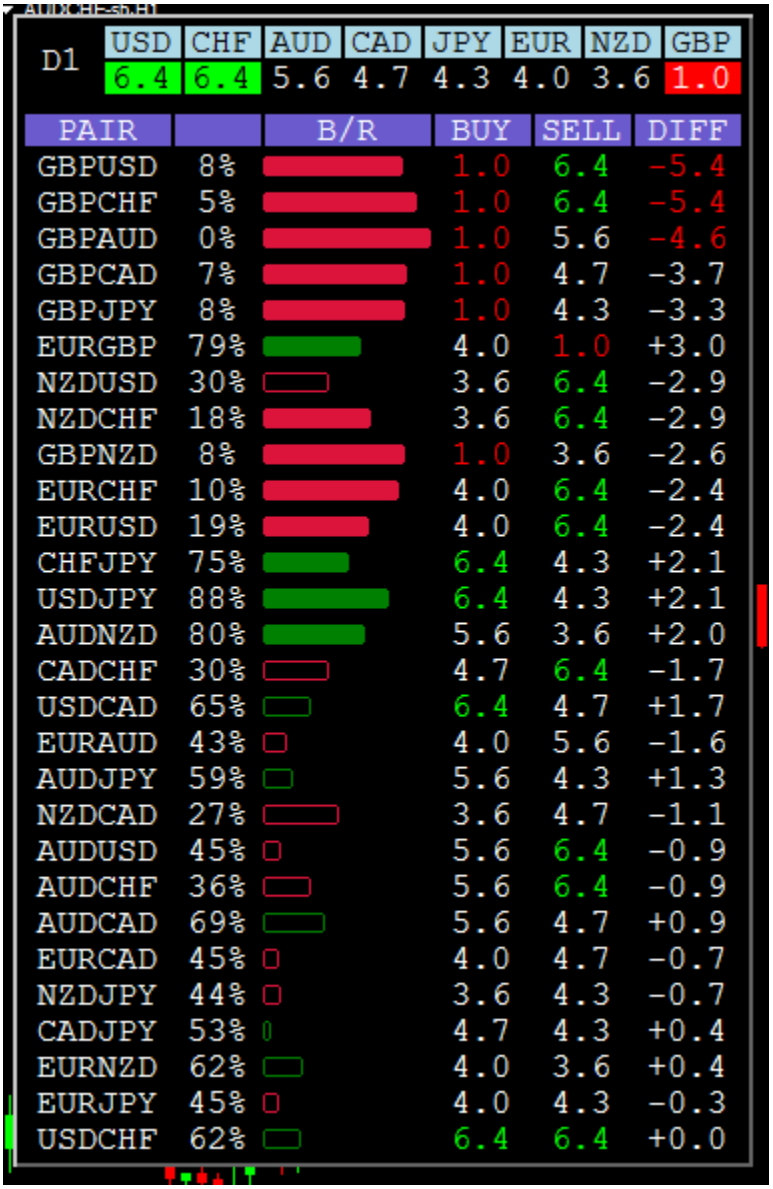
What does this mean? Bid ratio is where the current bid price is relative to the high and low. The start of the bar is 50%.

If the bar is colored red, it means the bid ratio is **below** 50%. The longer the bar, the closer to 0%. Below 25% is a solid red. Between 25 and 50% is hollow red.

If the bar is colored green, it means the bid ratio is **above** 50%. The longer the bar, the closer to 100%.

Above 75% is a solid green. Between 50-75% is hollow green.

This is quite hard to explain in words, so in this screenshot I have adjusted the spread column to display the actual bid ratio values. You can compare the actual bid values to how they look on the bar.



D1	USD	CHF	AUD	CAD	JPY	EUR	NZD	GBP
	6.4	6.4	5.6	4.7	4.3	4.0	3.6	1.0

PAIR		B/R	BUY	SELL	DIFF
GBPUSD	8%		1.0	6.4	-5.4
GBPCHF	5%		1.0	6.4	-5.4
GBPAUD	0%		1.0	5.6	-4.6
GBPCAD	7%		1.0	4.7	-3.7
GBPJPY	8%		1.0	4.3	-3.3
EURGBP	79%		4.0	1.0	+3.0
NZDUSD	30%		3.6	6.4	-2.9
NZDCHF	18%		3.6	6.4	-2.9
GBPNZD	8%		1.0	3.6	-2.6
EURCHF	10%		4.0	6.4	-2.4
EURUSD	19%		4.0	6.4	-2.4
CHFJPY	75%		6.4	4.3	+2.1
USDJPY	88%		6.4	4.3	+2.1
AUDNZD	80%		5.6	3.6	+2.0
CADCHF	30%		4.7	6.4	-1.7
USDCAD	65%		6.4	4.7	+1.7
EURAUD	43%		4.0	5.6	-1.6
AUDJPY	59%		5.6	4.3	+1.3
NZDCAD	27%		3.6	4.7	-1.1
AUDUSD	45%		5.6	6.4	-0.9
AUDCHF	36%		5.6	6.4	-0.9
AUDCAD	69%		5.6	4.7	+0.9
EURCAD	45%		4.0	4.7	-0.7
NZDJPY	44%		3.6	4.3	-0.7
CADJPY	53%		4.7	4.3	+0.4
EURNZD	62%		4.0	3.6	+0.4
EURJPY	45%		4.0	4.3	-0.3
USDCHF	62%		6.4	6.4	+0.0

The spreadsheets use the concept of **bid ratio:** where is the bid price relative to the day's high? If the bid happens to be the daily high, the bid ratio is 100%. If the bid is the daily low, the bid ratio is 0%.

This number then gets adjusted, factored and averaged but ultimately it is to do with how the 7 pairs are all moving.

Using the JPY example simply because you don't need to worry about base/quote (JPY is always the quote):
If all 7 pairs are close to the Daily Low, this implies JPY is very strong. Let's give them fictitious values:
USD**JPY** 5%
EUR**JPY** 7%
GBP**JPY** 4%
CHF**JPY** 2%
CAD**JPY** 4%
AUD**JPY** 3%
NZD**JPY** 5%
The average is 4.3%... a low number. As it is the quote currency, you need to turn it on its head and consider it 95.7%.

It would be fair to say that, in general terms (it all depends on how you are measuring correlation),

all the pairs are highly correlated and the currency is strong.

You remove some of the correlation (less consistency in the bid ratios) and you remove the strength:

USD**JPY** 12%
EUR**JPY** 83%
GBP**JPY** 16%
CHF**JPY** 23%
CAD**JPY** 76%
AUD**JPY** 54%
NZD**JPY** 93%
The average is 51%... a middle of the road number.

So not much correlation, and not much strength or weakness.

All you need is the formula:

(bid-low) / (high-low)

And the factoring:

0.00-0.03 = 0
0.03-0.10 = 1
0.10-0.25 = 2
0.25-0.40 = 3
0.40-0.50 = 4
0.50-0.60 = 5
0.60-0.75 = 6
0.75-0.90 = 7
0.90-0.97 = 8
0.97-1.00 = 9

Moving into the experimental realm: rolling time frames.

The formula for csDash is well known. It is based around the bid ratio i.e. where is the current Bid relative to the High and Low.

The "problem" with this is that the High-Low range is very narrow at the start of a new bar and slowly gets wider as time progresses.

Consider the classic D1 timeframe that the original Excel spreadsheet used.

Last Friday at 00:05 there was a 6.3 pip range on the D1 bar. This meant that 1 pip of movement was ~16% on bid ratio.

At 04:00 there was a 20.7 pip range on the D1 bar. This meant that 1 pip of movement was ~5% on bid ratio.

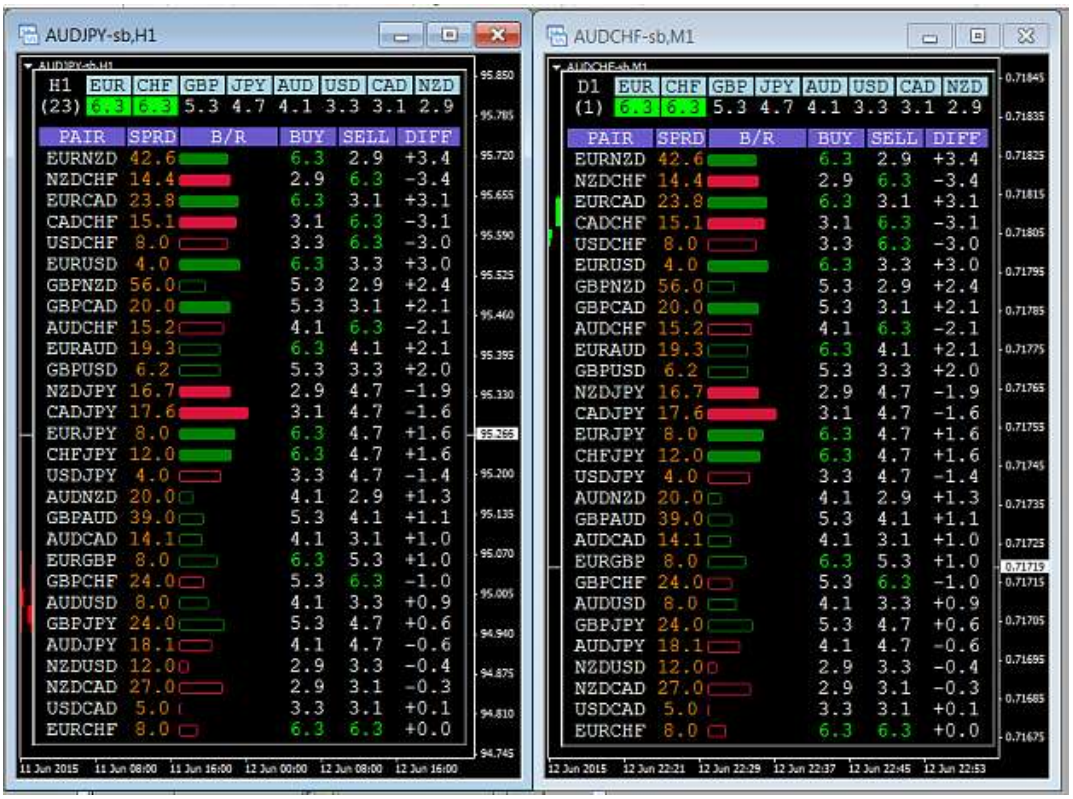
And at the close of the D1 bar on Friday, there was a 145.5 pip range on the D1 bar. This meant that 1 pip of movement was <1% on bid ratio

The net result of all of this: things go a bit crazy at the start of a new bar and slowly settle down as time progresses.

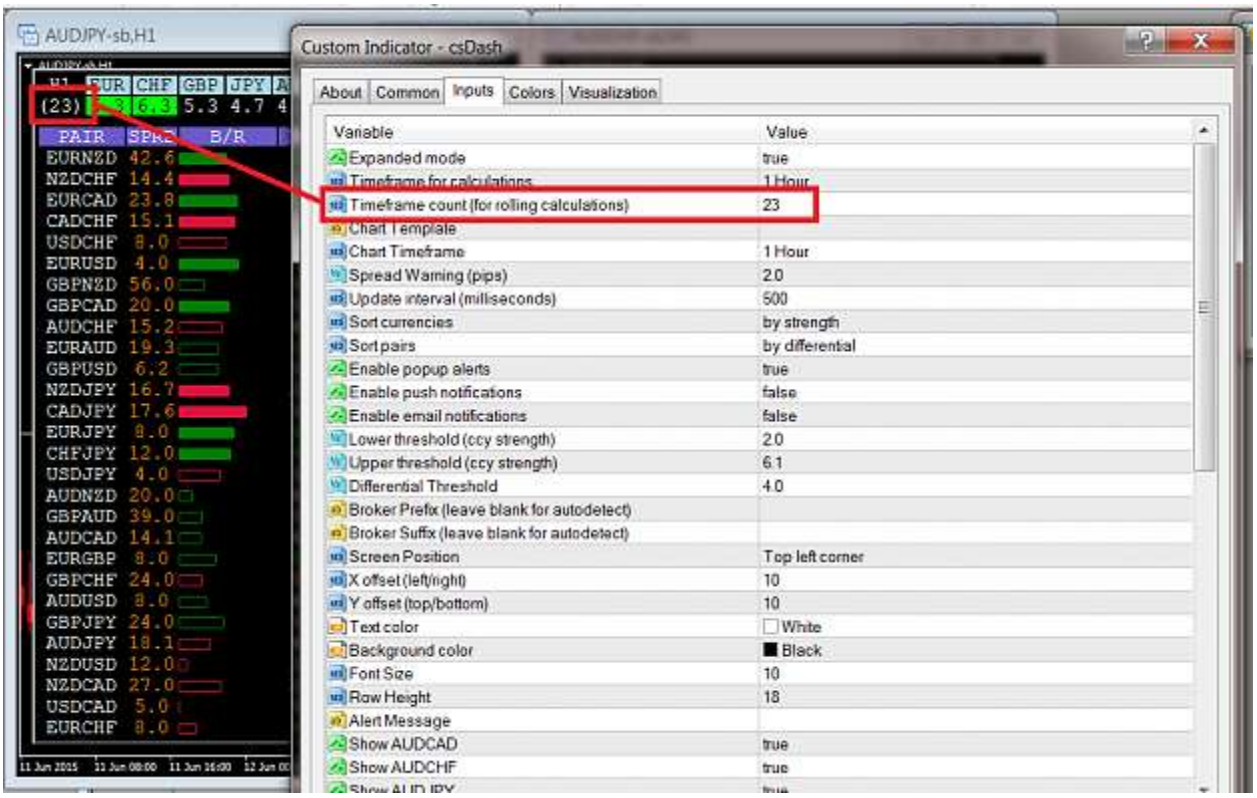
What the next version of csDash will let you do is roll timeframes.

So, instead of working from the D1 bar, you can work from the last 24 H1 bars. By doing so, the calculation is always done on 24 hours of data... not however long through the daily bar that time happens to be.

The results will be different during the day, but should show the same figures at the end of the day:
(for the observant ones who spotted that I only used 23 bars - it was a Friday and my broker doesn't have a 23:00 H1 bar but I wanted to show the same result. Normally, this would be set to 24)



If you don't like the idea of rolling timeframes, simply set the timeframe count to 1 and it will behave like it used to.



So, I've finally got around to implementing some experimental code to enhance the alerting process.

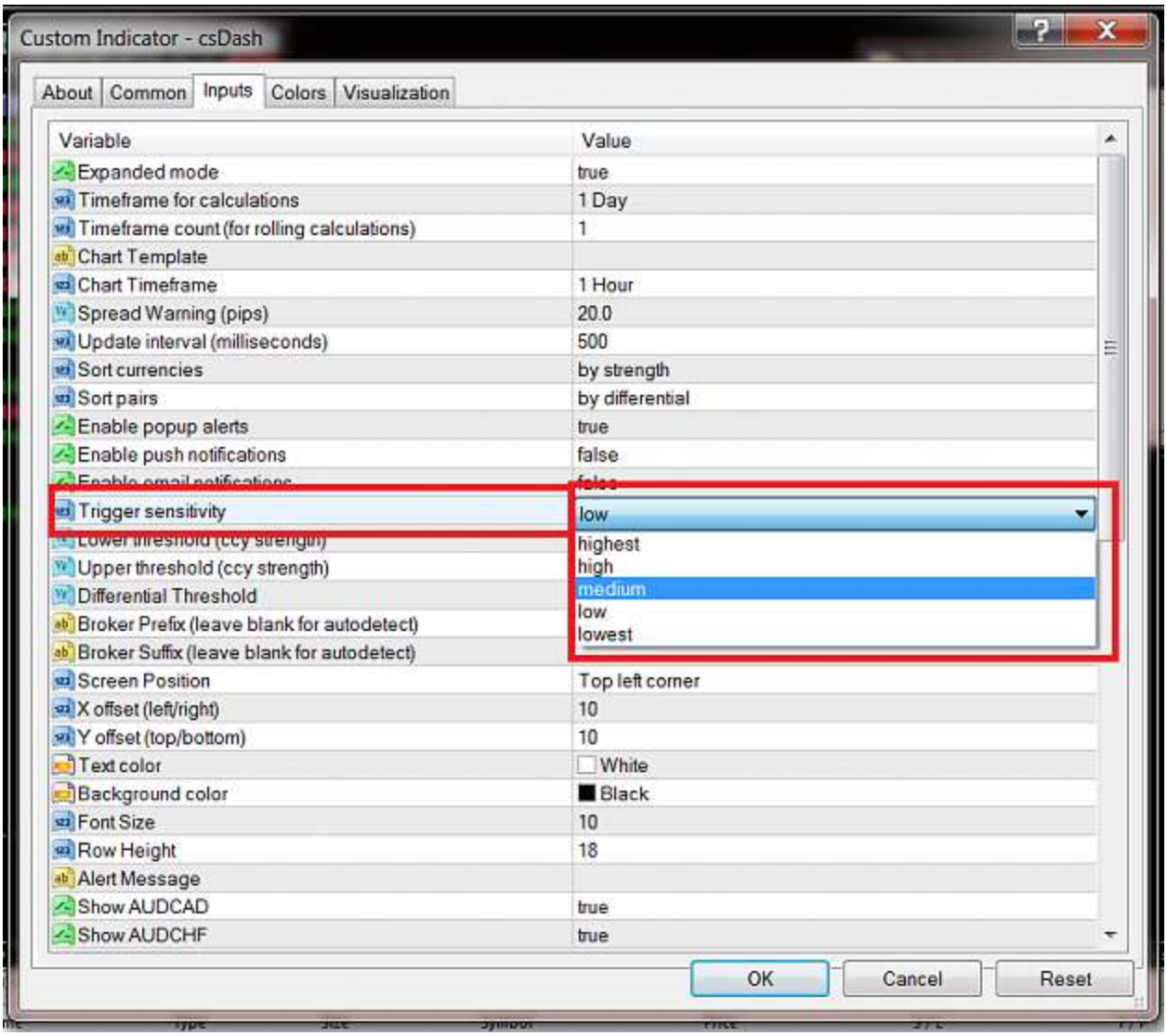
Why? The problem with alerts is that whenever a value hangs around an alert threshold (rather than moving clearly one way or the other) you get multiple triggers in close succession. These multiple alerts can create the impression that you are getting a really strong signal, but the reality is the opposite: you are getting a very marginal signal.

Consider the following example (differential threshold is 4.0):

3.8
3.9
4.0 ALERT!
4.1
4.0
3.9
4.0 ALERT!
3.9
4.0 ALERT!
3.9
3.9
4.0 ALERT!

csDash previously used a time delay of 1 minute between alerts. It is a crude mechanism based on the theory that things will move on and settle away from the threshold in the intervening minute. Doesn't always happen though.

I've now coded a new method to reduce the number of repeat alerts. The only user settings is Trigger Sensitivity:



At its highest setting, it works as per the example above. At its lowest setting, you will get very few **repeat** alerts (please note: sensitivity never affects the first alert).