

## ○ 「 history フォルダ中のシステム・ファイルを覗く (その1) 」

## ・アメンボです、

「¥history¥ブローカー名- Demo(I)、-Live 4」フォルダ中のシステム「4ファイル」、  
即ち「①symbols.sel、②symbols.raw、③symgrops.raw、④ticks.raw」の中身に  
興味を持ったことはありますか？

そう、土日に行うバックテスト時の「スプレッド」異常値に困惑したことのある諸兄は、  
既に一解決策として「symbols.sel」を書き換えるソフトやスクリプトを使ったことが  
あると思います。

(アメンボは当初、理由も対策も判らないまま「土日」をバックテストに使えず、苦労しました)

- ・小生は、書換ソフトやスクリプトの原理、逆って4ファイルの意味を理解したくて、  
以前から、何とかならないものかと考えていたのですが、幾つかの英語版サイトで、  
ヒントを見つけたので少し調べることにしました。  
しかし、MT4の内部仕様情報はとても少ないので、現状は部分的にしか理解していません。

- ・本稿では、「①symbols.sel、②symbols.raw、③symgrops.raw、④ticks.raw」について、  
確認できた内容を報告いたします。ただし、「未解決問題」が多々有り、  
更にアメンボの勘違い・間違いがあるかもしれませんが、ご容赦ください。  
また、メタ・クワーツ社はMT5の普及とバグ取りに精一杯でしょうから、  
MT4のシステムが大幅に変更されることは考え難いので、本情報は当面有効と考えます。

## &lt;同時掲載資料&gt;・・・ダウンロード用

- ・アメンボが動作確認に使用したスクリプト一式；  
「script\_for\_4files.zip」を解凍して試してみてください

---

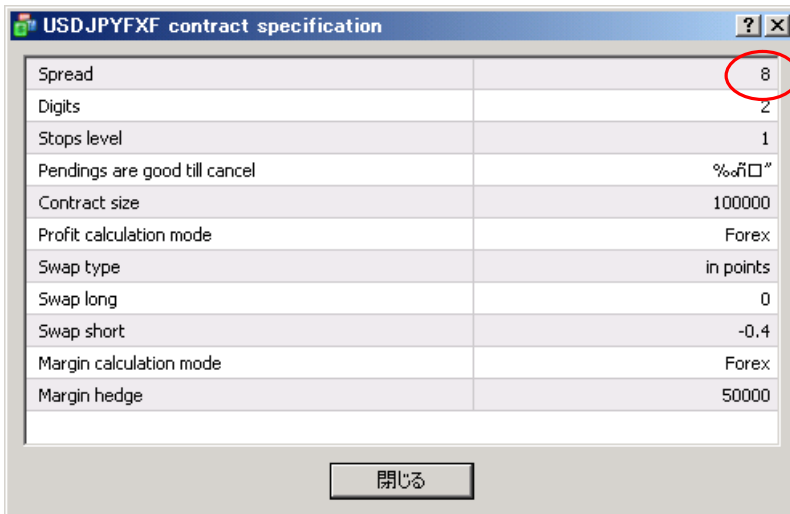
目次：	1. 土曜・日曜のスプレッド異常値	・・・ 2頁
	(1) アメンボの困惑・・・バックテストが出来ない！	
	(2) 良く知られた対策方法	
	2. 「¥history¥ブローカー名」ファイル中のシステム・4ファイル	・・・ 3頁
	(1) いつ更新されるのか？	
	(2) 役割は何か？・・・バックテストとの係わりは？！	
	3. 中身を調べてみる	・・・ 4頁～23頁
	(1) 現状の解析結果；4ファイルのMAP（地図）	
	(2) 解析手順と特記事項	
	4. 補足	・・・ 24頁～27頁
	(1) MT4をオフラインにする方法	
	(2) ライブ・トレード用の「contract specification」では、spreadが変！	
	(3) 次回（その2）の予定	

---

## 1. 土曜・日曜の спреッド異常値

### (1) アメンボの困惑・・・バックテストが出来ない！

・バックテストのとき、USDJPY で、こんな「Spread (スプレッド)」値を見た事ありませんか？



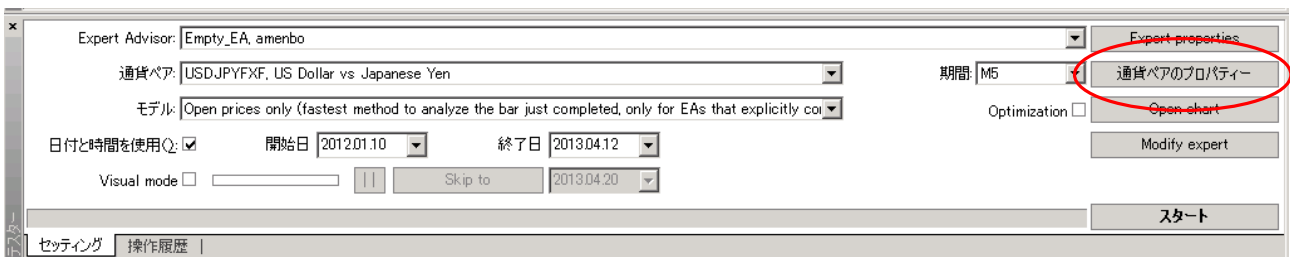
USDJPYFXF なのに、

！なんと「スプレッド=8」です。  
(いくら変動スプレッドと言っても)

既に原因と解決策を知っている諸兄が多いとは思いますが、敢えて復習しておきます。

お気づきの様に、これは土日にそのままバックテストをしようとすると起こります。

※補足；上図は、バックテスト時に [通貨ペアのプロパティ] をクリックすると表示されます。



- ・アメンボは、MT 4 で本格的にバックテスト・オプティマイズを始めた当初、月～金曜はそこそこの結果が出ていたEAが、土日曜になると成績が落ちてしまう理由が判らず困惑し、半年以上も土日にはバックテストをしないようにしていました。
- ・原因は上記に述べたように、土日曜のスプレッド異常値にあったようなのですが、判らずに随分と時間をロスしてしまいました。

### (2) 良く知られた対策方法

・アメンボが現時点で理解していて、現実的な「対策方法」は下記です。(過激な方法は省略)

#### －1. 土日曜にバックテストをしたいだけなら

ブローカーのサーバーが停止していない「月～金曜」に、MT 4 をオフラインにして、そのままバックテストをする。(土日曜にオンラインにするのは厳禁)

(MT 4 を簡単にオフラインにする方法の一例は「\*\*」頁を参照)

#### －2. スプレッドを一定値(任意値)にしてバックテストしたいなら

MT 4 をオフラインにしたうえで、良く知られた下記のツールでスプレッド値を設定する。

①SpreadChanger.exe

②SpreadGenerator.mq4・・・スクリプト

③TakeMySpread.exe・・・スプレッド、スワップ、Digits の書換(必要か否かは別として)

※上記のツールは、WEB 上で検索すれば直ぐにみつかります。

※何れにしても、アメンボは「バックテスト専用のMT 4」を設定しておくことを推奨します。

## 2. 「history」フォルダ中のシステム・4ファイル

- ・アメンボの悪い癖ができました！ そう、解析可能な限り、調べたくなかったです。

### (1) いつ更新されるのか？

<現状；アメンボの認識>

システム・4ファイル	更新タイミング	備考
symbols.sel	MT 4のシャットダウン時	シャットダウン・プロセスで更新
symbols.raw	MT 4のインストール時	以降、そのまま使われる ※1
symgroups.raw	MT 4のインストール時	以降、そのまま使われる ※1
ticks.raw	MT 4のシャットダウン時	シャットダウン・プロセスで更新

※1；チョット判らないことあり！

実は、「バックテスト」専用のMT 4として、  
初めて、2個目（2013）をインストールしてみたときに、  
1個目（2011年）のMT 4の同名ファイルも書き変わっていたのです！？  
（何故？？）

<上記判断の理由となった実験結果>



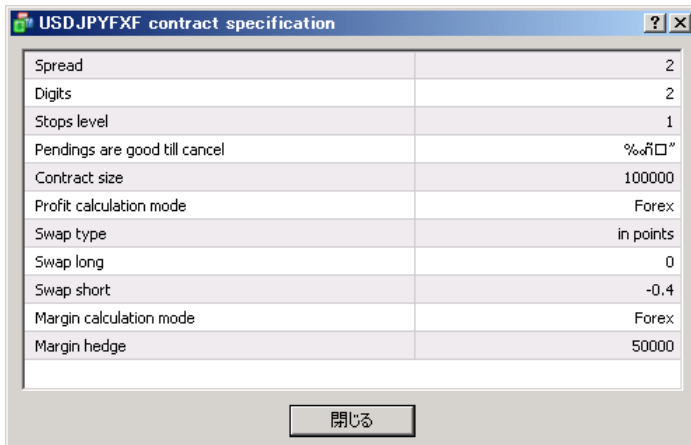
※バックテスト専用MT 4として、「2013/04/03」にダウンロード・インストール。

※上図は「2013/04/18；22時27分」にシャットダウンして、

「約20分後」に再起動したときの、ファイルの「更新日時」です。

### (2) 役割は何か？・・・バックテストとの係わりは？！

- ・アメンボの「資料段階」調査では「バックテスト」に必要なデータが含まれているようです。  
「ようです」と表現したのは、アメンボは、これから確認しようとしているからです。
- ・資料を調べた限りでは、実績があるものと判断中です



- ・要は、左図に表示される contract specification (契約仕様) ですね。  
( たぶん、間違いないと推測します )

### 3. 中身を調べてみる

- ・先ず、謙譲の解析結果要約（4ファイルのMAP）を示し、次に、解析して行った手順と特記事項を解説します。

#### (1) 現状の解析結果；4ファイルのMAP（地図）

##### － 1. < symbols.sel >のマップ

	セクター内オフセット(バイト)		バイト数	型	内容	MT4表現型	特記・他(内容)	
	10進	16進						
							1セクター=128(0x80)バイト	
	-	-	4	Int	Version NO	LONG_VALUE	セクターのカウン트에含めず	
第1セクター	1	0	0	12	Char	Symbol Short Name	String	「¥0`ヌル」終端
	2	12	0C	4	Int	Digts	LONG_VALUE	
	3	16	10	4	Int	1とか 263	LONG_VALUE	アメンボ経験値
	4	20	14	4	Int	1	LONG_VALUE	アメンボ経験値
	5			4	Int	4	LONG_VALUE	アメンボ経験値
	6			4	Int	0	LONG_VALUE	(Int)ギャップ
	7	32	20	8	Double	Point	DOUBLE_VALUE	
	8	40	28	4	Int	0	LONG_VALUE	(Int)ギャップ
	9	44		4	Int	0	LONG_VALUE	(Int)ギャップ
	10	48	30	4	Int	1	LONG_VALUE	アメンボ経験値
	11	52	34	4	Int	256	LONG_VALUE	アメンボ経験値
	12	56	38	4	Int	TimeCurrent()	LONG_VALUE (datetime)	シャットダウン時のデータ
	13	60	3C	4	Int	0	LONG_VALUE	(Int)ギャップ
	14	64	40	8	Double	Bid	DOUBLE_VALUE	シャットダウン時のデータ
	15			8	Double	Ask	DOUBLE_VALUE	シャットダウン時のデータ
	16	80	50	8	Double	High[0]	DOUBLE_VALUE	日足データ;シャットダウン時
	17			8	Double	Low[0]	DOUBLE_VALUE	日足データ;シャットダウン時
	18		60	8	Double	0	DOUBLE_VALUE	(Double)ギャップ
	19			8	Double	0	DOUBLE_VALUE	(Double)ギャップ
	20	112	70	8	Double	Bid	DOUBLE_VALUE	シャットダウン時のデータ
	21			8	Double	Ask	DOUBLE_VALUE	シャットダウン時のデータ
第2セクター	1	0	0	12	Char	Symbol Short Name	String	「¥0`ヌル」終端
	2	12	0C	4	Int	Digts	LONG_VALUE	
	3		10	4	Int	27	LONG_VALUE	
	4			4	Int	1	LONG_VALUE	

- ・ファイルの先頭には、「4バイト」の「バージョンNo」があります、その後には、1為替ペアにつき、「128バイト（16進で80）」の記述が続きます。  
( バイナリ・ファイルなので、「文字列」以外はフォーマットが判らないと読み出せません )
- ・「アメンボ経験値」とした部分は、内容不明の部分です。
- ・「16進」で表現すると、重要データのオフセット値が、とても「切りの良い値」になります、「0値」はそのための調整用では？と考えて「(Double)ギャップ」としました。

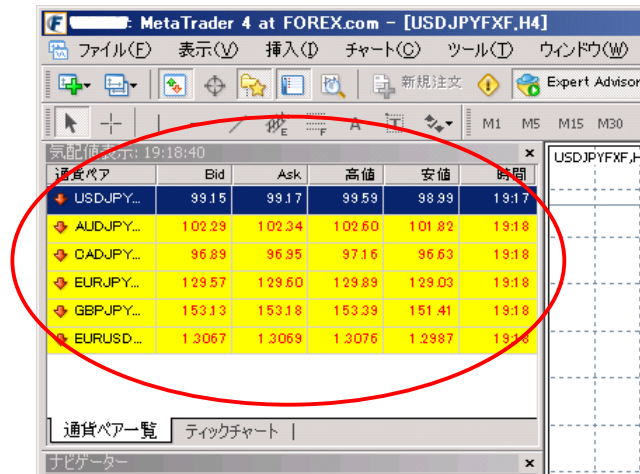
特記事項 ; 「Market Watch ウィンドウ」と「symbols.sel」の関係

<注意>

- ・本内容は「2013.04 現在」、アメンボが確認作業を経て、ほぼ大丈夫であろうと考える内容です、「万が一、間違いの可能性もあります」。(その場合はご容赦)

Market Watch ウィンドウの役割 ; (再確認)

※「Market Watch (マーケット・ウォッチ) ウィンドウ」とは、例えば下記のことです。(図1)



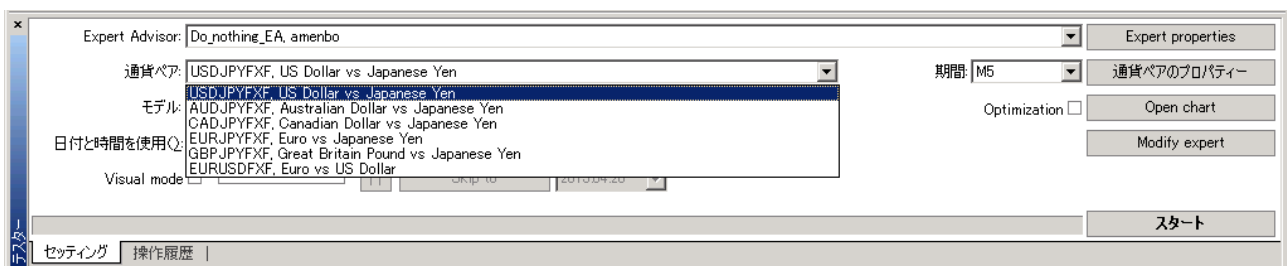
「Market Watch (マーケット・ウォッチ) ウィンドウ」と「symbols.sel」

- ・Market Watch ウィンドウの内容は、MT 4のシャット・ダウン時に、「symbols.sel」内容に反映されます。

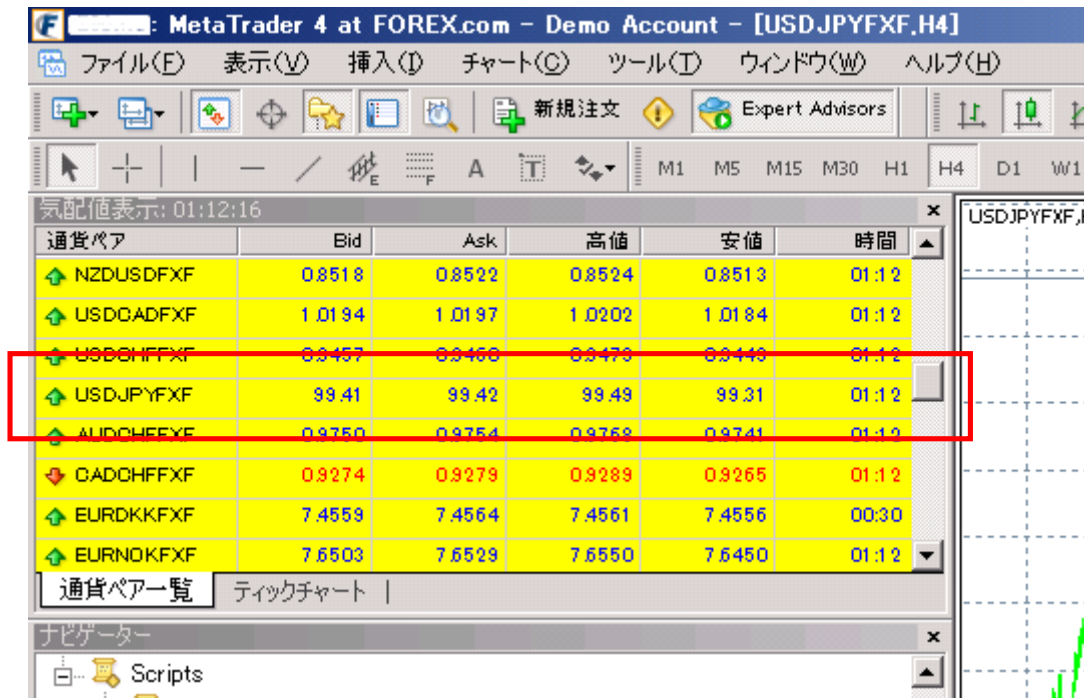
Market Watch 内容	反映ファイル	備考
サーバー時間の表示	symbols.sel	
サーバー配信レート (Bid、Ask) 表示	symbols.sel	
その日の「高値、安値」表示	symbols.sel	日足の「高値、安値」です
ティック・データ表示	?不明	ticks.raw に関係するかも??

「Market Watch (マーケット・ウォッチ) ウィンドウ」と「バックテスト可能な通貨ペア」

- ・「Market Watch ウィンドウ」と「バックテスト可能な通貨ペア」は、一致します。例えば、上記「図1」の場合には、バックテスト可能な「通貨ペア」は、下記ようになります。・(Market Watch ウィンドウの為替ペアと同じ)



※実験結果； 「-1。」の証明・・・「symbols.sel」書き出しスクリプトは「\*\*頁」参照  
 ・為替ペア「USDJPYFXF」の、MT 4 シャット・ダウン直前「Market Watch ウィンドウ」と再起動後の「symbols.sel」内容を比較した。・・・内容一致を確認



※実際に、画像1の状態MT 4をシャットダウンし(1分遅れた)、約10分経過後に「symbols.sel」の内容(USDJPYFXF部分)を書出した結果を下記に示します。(¥experts¥logs¥日付.logの内容)

```

01:23:36 symbols_sel_content_of_pair USDJPYFXF, H4: loaded successfully
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4 inputs: Symbol_name="USDJPYFXF";
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Version NO= 400
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: -----START-----
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: ShortName= USDJPYFXF
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Digits= 2
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Point= 0.01
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4:
    TimeCurrent() in symbols.sel_file= 1366938735
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4:
    Time_in_symbols.sel_file:2013年4月26日1時12分15秒
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4:
    current server time:2013年4月26日1時23分31秒
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4:
    PC time:2013年4月26日1時23分37秒
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Bid_1= 99.41 || ask_1= 99.42
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: High[0]= 99.49 || Low[0]= 99.31
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Bid_2= 99.41 || ask_2= 99.42
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: ----Gap List----
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Gap1~Gap4= 263;1;4;0
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Gap5~Gap8= 0;0;0;256
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Gap6= 0
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: Gap7, Gap8= 0;0
01:23:37 symbols_sel_content_of_pair USDJPYFXF, H4: -----END-----
  
```

- 2. < symbols.raw >のマップ

	セクター内オフセット(バイト)		バイト数	型	内容	MT4表現型	備考	
	10進	16進						
							1セクター=1396バイト	
第1セクター	1	0	0	12	char	Symbol Short Name	string	「¥0`ヌル」終端
	2	12	12	64	char	Description	string	「¥0`ヌル」終端
		76	4C					
	3	1680	690	8	double	Swap_Long Value	DOUBLE_VALUE	
	4	1688	698	8	double	Swap_Short Value	DOUBLE_VALUE	
			6a0					
	5	1704	6a8	8	double	Contract Size	DOUBLE_VALUE	
			6b0					
6	1728	6c0	1	char	Stop Level	CHAR_VALUE		
		6c1						
第2セクター	1	0	0	12	char	Symbol Short Name	string	「¥0`ヌル」終端
	2	12	12	64	char	Description	string	「¥0`ヌル」終端

※中身は、殆ど判っていません。

- 3. < symgroups.raw >のマップ

	セクター内オフセット(バイト)		バイト数	型	内容	MT4表現型	計算式・他(内容)
	10進	16進					
			max値				1ブロック=80(0x50)バイト
1	0	0	16	char	FXのコース名??	String	文字としては15バイト可能、16バイト目はヌル文字
2	12	0C	64	char	コースの概要説明か?	String	文字としては63バイト可能、64バイト目はヌル文字
1	0	0	16	char	FXのコース名??	String	文字としては15バイト可能、16バイト目はヌル文字
2	12	0C	64	char	コースの概要説明か?	String	文字としては63バイト可能、64バイト目はヌル文字
1	0	0	16	char	FXのコース名??	String	文字としては15バイト可能、16バイト目はヌル文字
2	12	0C	64	char	コースの概要説明か?	String	文字としては63バイト可能、64バイト目はヌル文字

※殆どが、テキスト・データの様に見えます。でも、それ以外のデータがあるかは不明。

- 4. < ticks.raw >のマップ

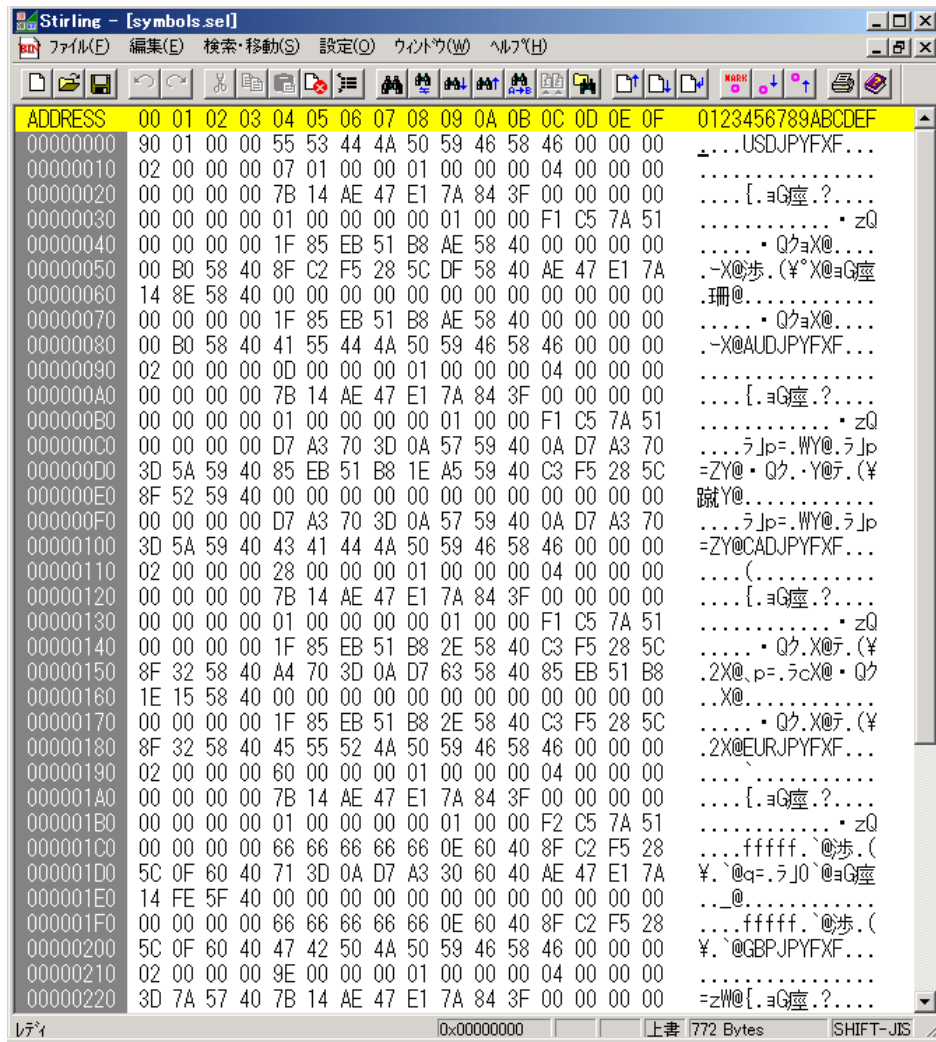
	セクター内オフセット(バイト)		バイト数	型	内容	MT4表現型	計算式・他(内容)
	10進	16進					
第1セクター	1	0	12	Char	Symbol Short Name	String	「¥0`ヌル」終端
	2	12	??	??	??		

※ 「Symbol Short Name」以外は何も判っていません。  
 もしかすると、「ティック・チャート」のデータと関係が有りそうだと考えています??

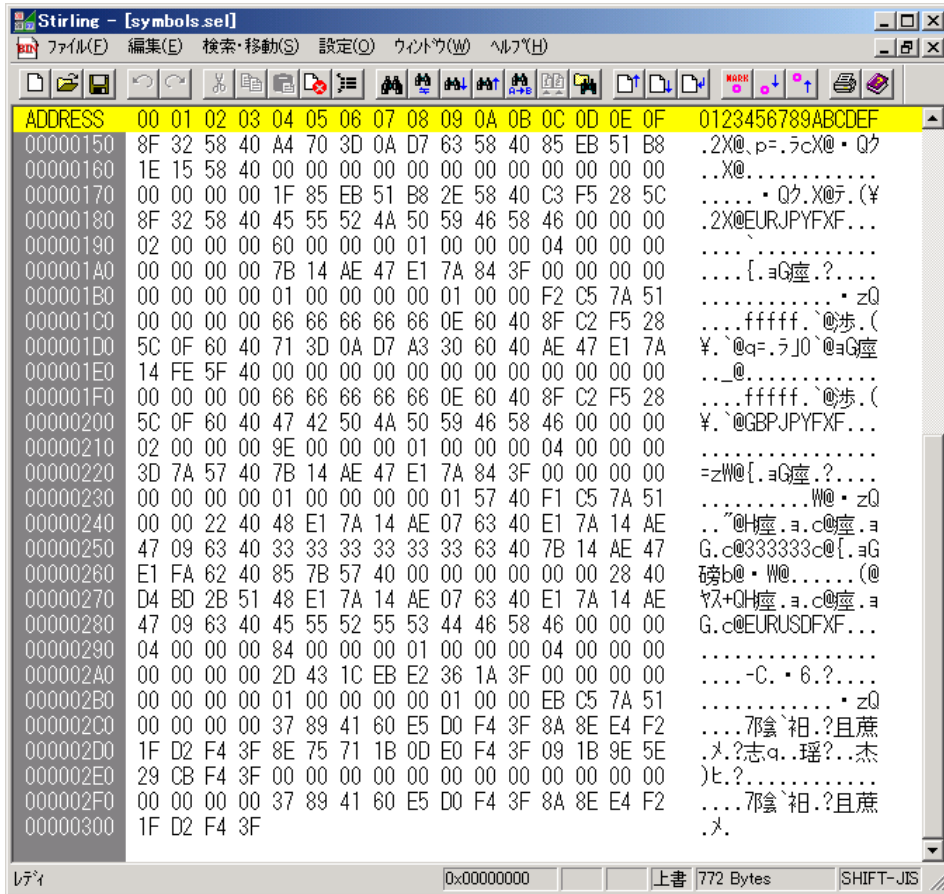
(2) 解析手順と特記事項

A. 16進エディタで覗く

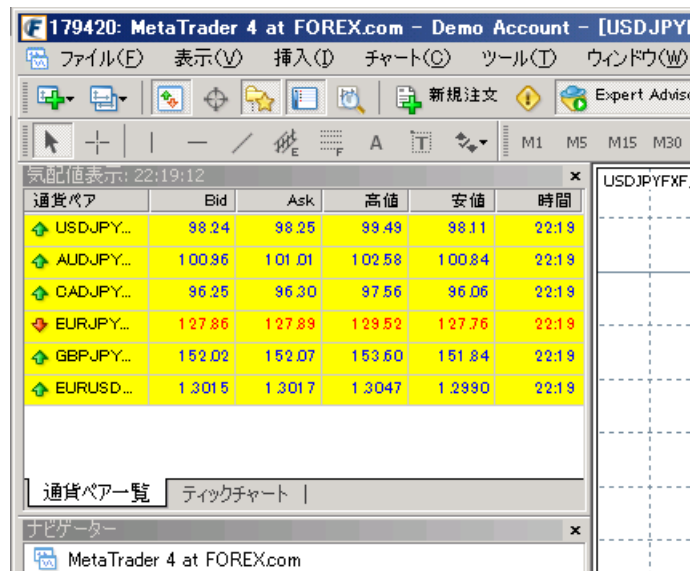
- 1. < symbols.sel >







※比較用に、この時のMT 4の「Market Watch ウィンドウ」を示しておきます。



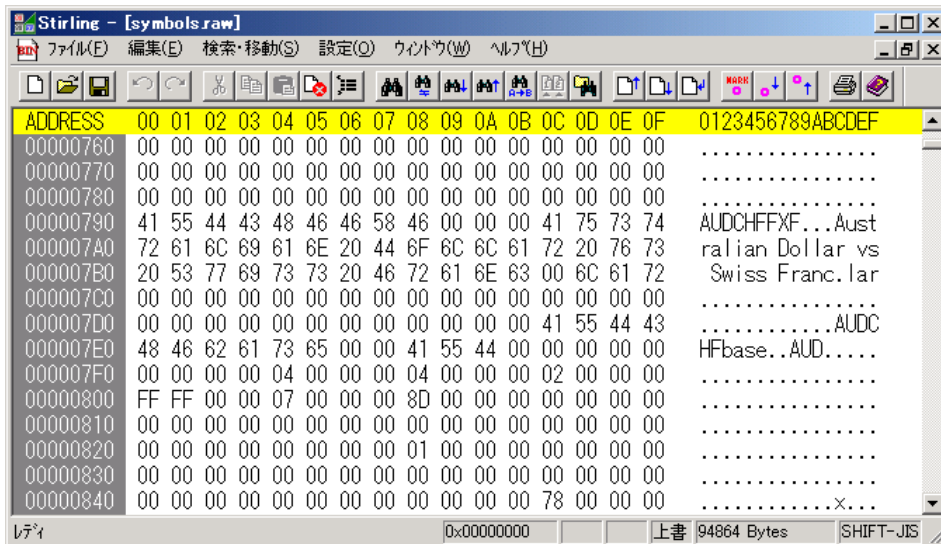
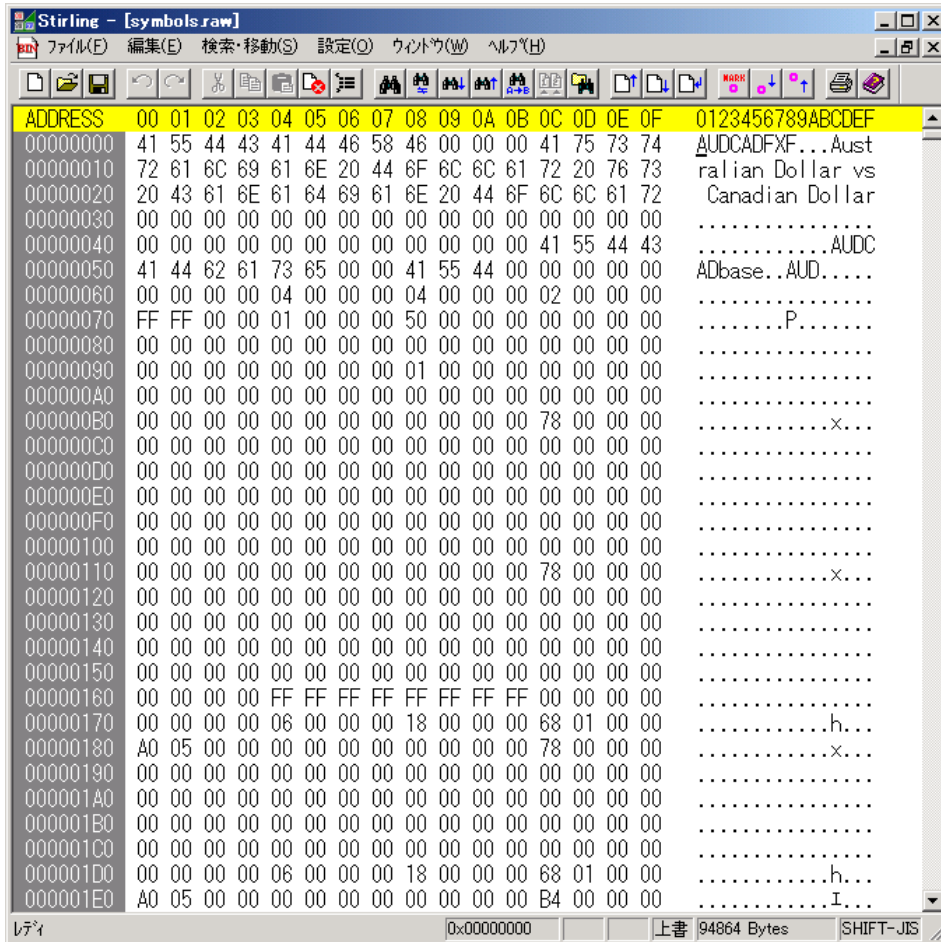
※比較して即、判ること；

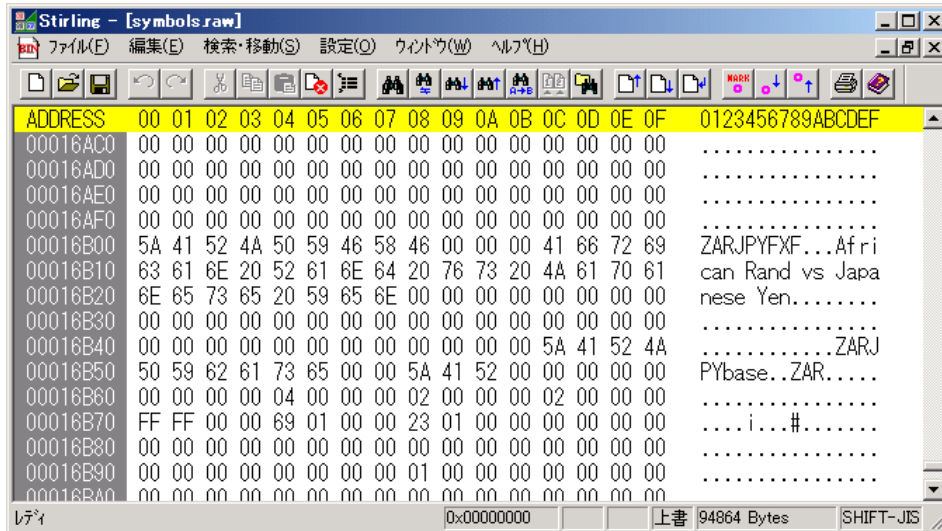
- ・「symbols.sel」には、「Market Watch ウィンドウ」と同じ順番で、通貨ペアのデータが書き込まれています。(収録される通貨ペアの数も同じ)

※16進エディタとしては、一般的な「Stirling (フリーウェア)」を使用しました。

※内容のうち、「文字列」データは「¥0」(ヌル)で終端されるので、読めるのですが、他のデータは隙間無く詰め込まれるので、フォーマットが判らないと読めません。

- 2. < symbols.raw >

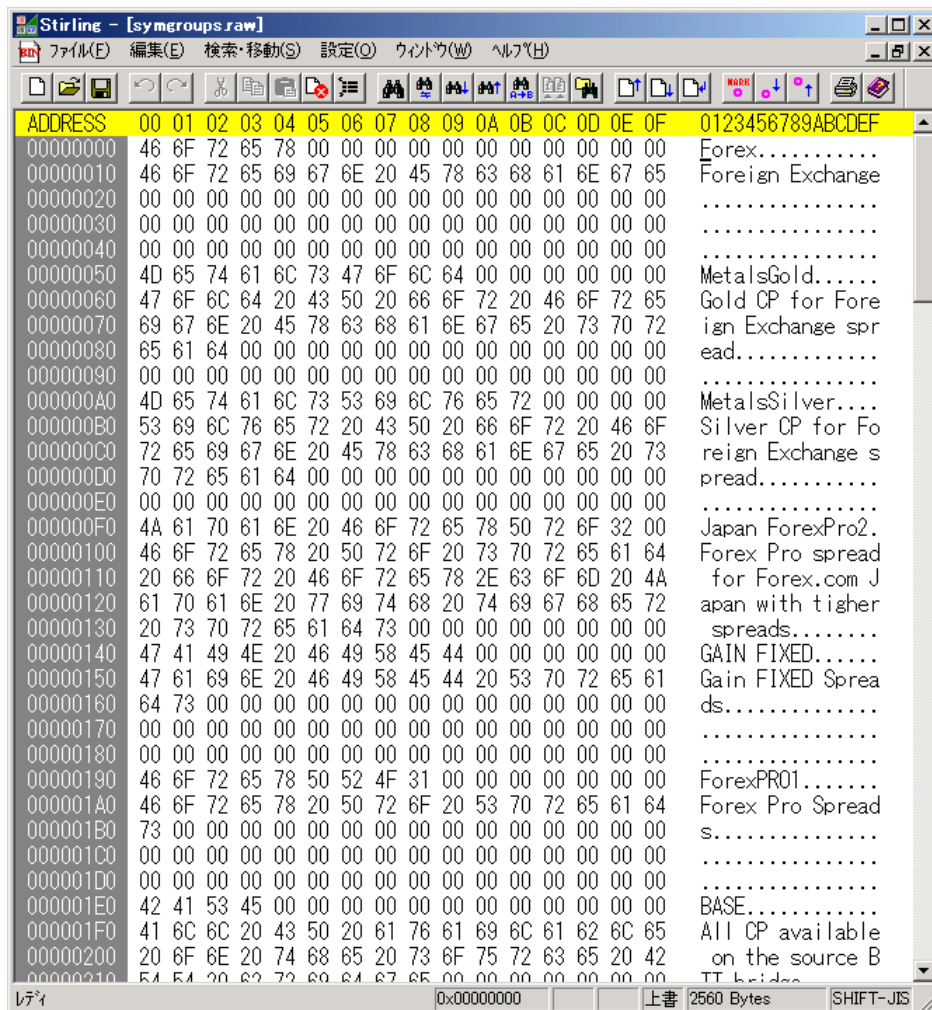




※ブローカーが提供している「全ての通貨ペア」の名称が見えます。

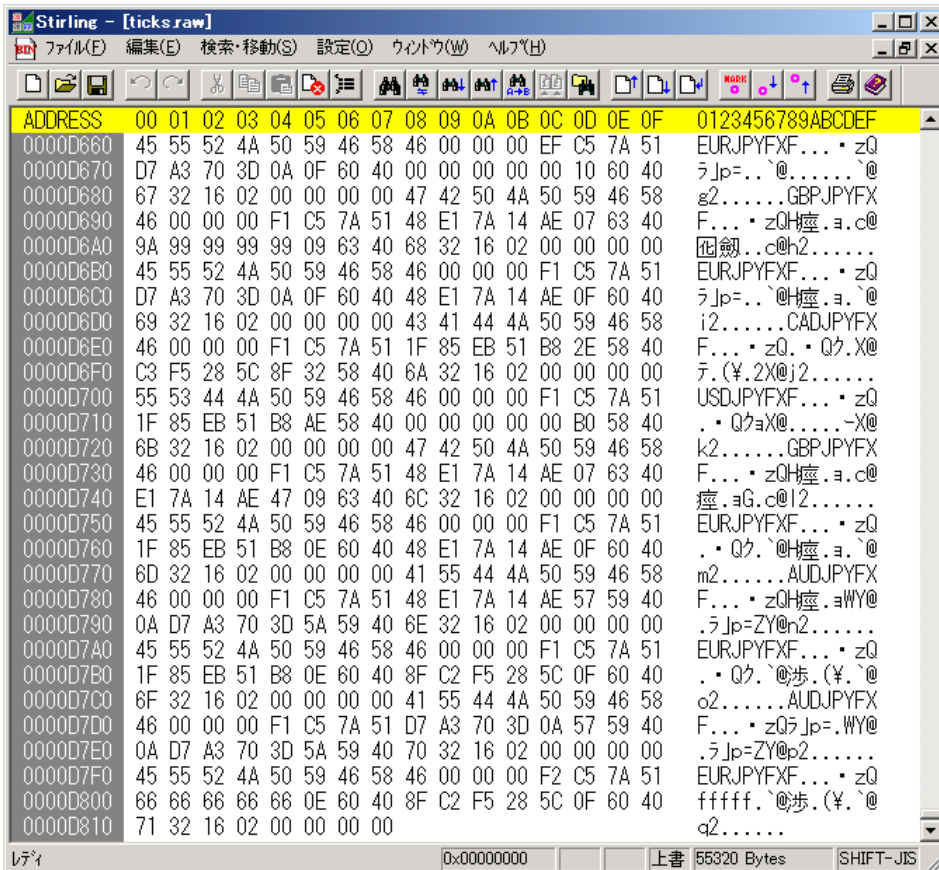
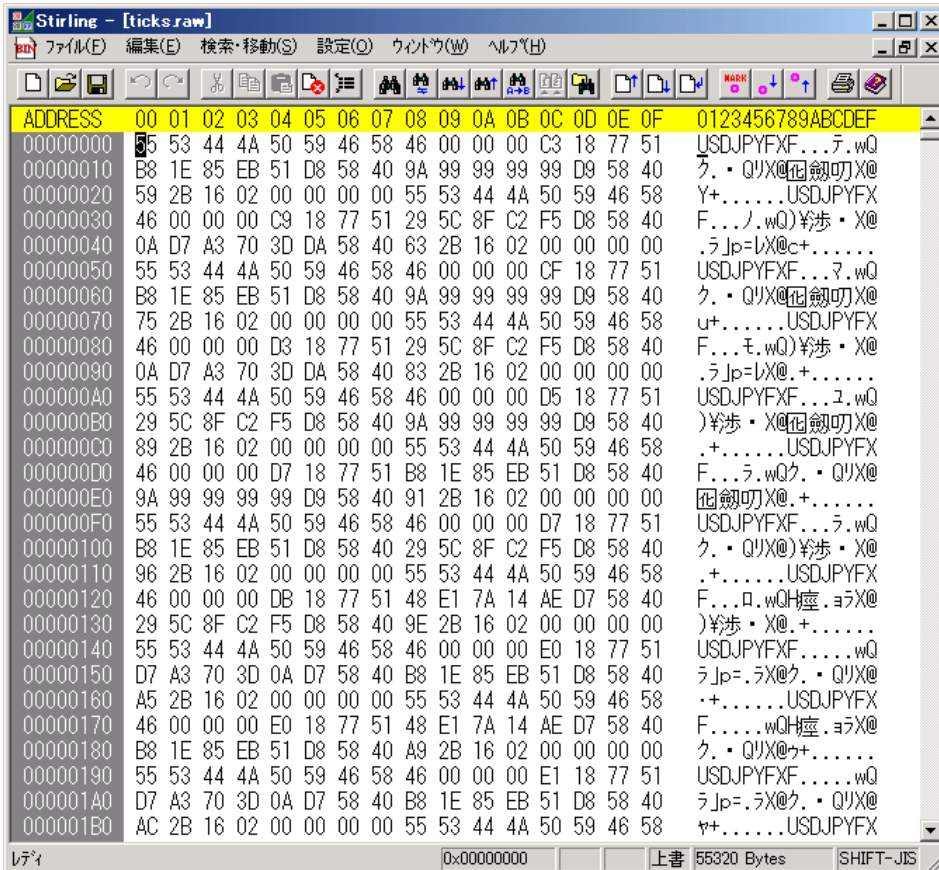
※ただ、やけに中身の密度が低いように見えます。

－ 3. < symgroups.raw >



※覗いた範囲では、「文字列」のみが在るように見えます。

- 4. < ticks.raw >



※現状、通貨ペアのショート・ネーム以外は、全くわかりません。

## B. フレームをプリントしてみる

- ・英語のWEBサイトからの情報、および16進エディタで覗いた内容から判明した範囲で、「第1セクターのフレーム確認内容」と、「任意セクター（通貨ペア）の内容」を、「¥experts¥logs¥日付.log」に書出すスクリプトを作成してみました。

	第1セクター書き出し	任意通貨ペア書き出し	備考
symbols.sel	symbols_sel_Frame.mq4	symbols_sel_content_of_pair.mq4	何箇所かは不明
symbols.raw	symbols_raw_Frame.mq4	symbols_raw_content_of_pair.mq4	一部のみ判明
symgroups.raw	symgroups_raw_Frame.mq4	情報未入手で、作れず	でも、全部文字列
ticks.raw	ticks_raw_Frame.mq4	情報未入手で、作れず	判らない！！

- ・ここ「B」ではフレームを、次の「C」では内容をプリントする例を示します。

### - 1. < symbols.sel >

①コード (symbols\_sel\_Frame.mq4)

```
//+-----+
//|                                     |
//|                                     |
//|                                     |
//|                                     |
//+-----+
#property copyright "amenbo"
#property link      "泉の森の弁財天池"

int start()
{
    string SymbolsList[];
    int version;

    string SymbolsFileName= "symbols.sel";
    int Offset=128;
    int SymbolsNumber;
    int hFile = FileOpenHistory(SymbolsFileName, FILE_BIN|FILE_READ);

    Print("start-----");

    FileSeek(hFile, 0, SEEK_SET);
    version=FileReadInteger(hFile, LONG_VALUE);//この処理でカーソルは「4バイト」移動する
    Print("version NO= ", version);

    SymbolsNumber=((FileSize(hFile)-4)/128);//ファイル先頭の「4バイト」は計算から除外する

    for(int i=1;i<=SymbolsNumber;i++)
    {
        SymbolsList[i] = FileReadString(hFile, 12);
        Print("SymbolsList[", i, "]= ", SymbolsList[i]);
        FileSeek(hFile, (Offset-12), SEEK_CUR);
    }

    Print("end-----");

    FileClose(hFile);

    PlaySound("alert2.wav");

    return(0);
}
//+-----+
```

②結果 (¥experts¥logs¥日付.log)

23:58:38 symbols\_sel\_Frame USDJPYFXF, H4: loaded successfully

```

23:58:38 symbols_sel_Frame USDJPYFXF, H4: start-----
23:58:38 symbols_sel_Frame USDJPYFXF, H4: version NO= 400
23:58:38 symbols_sel_Frame USDJPYFXF, H4: SymbolsList[1]= USDJPYFXF
23:58:38 symbols_sel_Frame USDJPYFXF, H4: SymbolsList[2]= AUDJPYFXF
23:58:38 symbols_sel_Frame USDJPYFXF, H4: SymbolsList[3]= CADJPYFXF
23:58:38 symbols_sel_Frame USDJPYFXF, H4: SymbolsList[4]= EURJPYFXF
23:58:38 symbols_sel_Frame USDJPYFXF, H4: SymbolsList[5]= GBPJPYFXF
23:58:38 symbols_sel_Frame USDJPYFXF, H4: SymbolsList[6]= EURUSDFXF
23:58:38 symbols_sel_Frame USDJPYFXF, H4: end-----

```

## - 2. < symbols.raw >

①コード (symbols\_raw\_Frame.mq4)

```

//-----+-----+
//|                                     | symbols_raw_Frame.mq4 |
//|                                     |      amenbo          |
//|                                     |      泉の森の弁財天池 |
//-----+-----+
#property copyright "amenbo"
#property link      "泉の森の弁財天池"

int start()
{
    string SymbolsList[];
    string SymbolsDescription[];
    int position;

    string SymbolsFileName= "symbols.raw";
    int Offset=1936;// 1通貨ペア分のサイズ
    int SymbolsNumber;
    int hFile = FileOpenHistory(SymbolsFileName, FILE_BIN|FILE_READ);

    SymbolsNumber=(FileSize(hFile)/Offset);
    FileSeek(hFile, 0, SEEK_SET);

    Print("start-----");

    for(int i=1;i<=SymbolsNumber;i++)
    {
        SymbolsList[i] = FileReadString(hFile, 12);//カーソルは「12バイト」動く
        SymbolsDescription[i]=FileReadString(hFile, 64);
        Print("SymbolsList[" , i, "] = " , SymbolsList[i]);
        Print("SymbolsDescription[" , i, "] = " , SymbolsDescription[i]);
        FileSeek(hFile, (Offset-(12+64)), SEEK_CUR);
    }

    Print("end-----");

    FileClose(hFile);

    PlaySound("alert2.wav");

    return(0);
}
//-----+-----+

```

②結果 (¥experts¥logs¥日付.log)

```

23:59:56 symbols_raw_Frame USDJPYFXF, H4: loaded successfully
23:59:56 symbols_raw_Frame USDJPYFXF, H4: start-----
23:59:56 symbols_raw_Frame USDJPYFXF, H4: SymbolsList[1]= AUDCADFXF
23:59:56 symbols_raw_Frame USDJPYFXF, H4: SymbolsDescription[1]= Australian Dollar vs Canadian Dollar
23:59:56 symbols_raw_Frame USDJPYFXF, H4: SymbolsList[2]= AUDCHFXXF
23:59:56 symbols_raw_Frame USDJPYFXF, H4: SymbolsDescription[2]= Australian Dollar vs Swiss Franc
23:59:56 symbols_raw_Frame USDJPYFXF, H4: SymbolsList[3]= AUDJPYFXF

```

23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[3]= Australian Dollar vs Japanese Yen  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[4]= AUDNZDFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[4]= Australian Dollar vs New Zealand Dollar  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[5]= AUDUSDFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[5]= Australian Dollar vs US Dollar  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[6]= CADCHFFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[6]= Canadian Dollar vs Swiss Franc  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[7]= CADJPYFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[7]= Canadian Dollar vs Japanese Yen  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[8]= CHFJPYFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[8]= Swiss Franc vs Japanese Yen  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[9]= EURAUDFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[9]= Euro vs Australian Dollar  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[10]= EURCADFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[10]= Euro vs Canadian Dollar  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[11]= EURCHFFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[11]= Euro vs Swiss Franc  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[12]= EURCZKFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[12]= Euro vs Czech Koruna  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[13]= EURDKKFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[13]= Euro vs Danish Krone  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[14]= EURGBPFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[14]= Euro vs Great Britain Pound  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[15]= EURHUFFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[15]= Euro vs Hungarian Forints  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[16]= EURJPYFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[16]= Euro vs Japanese Yen  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[17]= EURNOKFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[17]= Euro vs Norwegian Krone  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[18]= EURNZDFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[18]= Euro vs New Zealand Dollar  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[19]= EURPLNFXF  
 . . . . . 途中は正暦 . . . . .  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[35]= USDCHFFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[35]= US Dollar vs Swiss Franc  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[36]= USDCNHFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[36]= US Dollar vs Chinese Yuan  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[37]= USDCZKFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[37]= US Dollar vs Czech Koruna  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[38]= USDDKKFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[38]= US Dollar vs Danish Krone  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[39]= USDHUFFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[39]= US Dollar vs Hungarian Forints  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[40]= USDJPYFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[40]= US Dollar vs Japanese Yen  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[41]= USDMXNFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[41]= US Dollar vs Mexican Peso  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[42]= USDNOKFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[42]= US Dollar vs Norwegian Krone  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[43]= USDPLNFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[43]= US Dollar vs Polish Zloty  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[44]= USDRUBFXF  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsDescription[44]= US Dollar vs Russian Ruble  
 23:59:56 symbols\_raw\_Frame USDJPYFXF,H4: SymbolsList[45]= USDSEKFXF



## - 3. &lt; symgroups.raw &gt;

①コード (symgroups\_raw\_Frame.mq4)

```

//+-----+
//|                                     symgroups_raw_Frame.mq4 |
//|                                     amenbo |
//|                                     泉の森の弁財天池 |
//+-----+
#property copyright "amenbo"
#property link      "泉の森の弁財天池"
//
int start()
{
    string string_01[], string_02[];
    string SymbolsFileName= "symgroups.raw";
    int Offset=0x50;//0d80 バイト
    int SymbolsNumber;

    int hFile = FileOpenHistory(SymbolsFileName, FILE_BIN|FILE_READ);

    Print("start-----");

    FileSeek(hFile, 0, SEEK_SET);
    SymbolsNumber=(FileSize(hFile)/Offset);

    Print("為替ペア数 (SymbolsNumber) は ",SymbolsNumber," です。");

    for(int i=0;i<=(SymbolsNumber-2);i++)
    {
        FileSeek(hFile, (Offset*i), SEEK_SET);

        int cur_01=FileTell(hFile);
        string_01[i+1]= FileReadString(hFile, 16);
        Print("cur_01= ",cur_01," || string_01[",i+1,"]= ",string_01[i+1]);

        int cur_02=FileTell(hFile);
        string_02[i+1]= FileReadString(hFile, 64);
        Print("cur_02= ",cur_02," || string_02[",i+1,"]= ",string_02[i+1]);
    }

    Print("end-----");

    FileClose(hFile);

    PlaySound("alert2.wav");

    return(0);
}
//+-----+

```

②結果 (¥experts¥logs¥日付.log)

```

00:01:35 symgroups_raw_Frame USDJPYFXF,H4: loaded successfully
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: start-----
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: 為替ペア数 (SymbolsNumber) は 32 です。
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_01= 0 || string_01[1]= Forex
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_02= 16 || string_02[1]= Foreign Exchange
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_01= 80 || string_01[2]= MetalsGold
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_02= 96 || string_02[2]= Gold CP for Foreign Exchange spread
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_01= 160 || string_01[3]= MetalsSilver
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_02= 176 || string_02[3]= Silver CP for Foreign Exchange spread
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_01= 240 || string_01[4]= Japan ForexPro2
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_02= 256 || string_02[4]= Forex Pro spread for Forex.com Japan
with tigher spreads
00:01:35 symgroups_raw_Frame USDJPYFXF,H4: cur_01= 320 || string_01[5]= GAIN FIXED

```



```

00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 336 || string_02[5]= Gain FIXED Spreads
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 400 || string_01[6]= ForexPR01
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 416 || string_02[6]= Forex Pro Spreads
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 480 || string_01[7]= BASE
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 496 || string_02[7]= All CP available on the source BTT bridge
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 560 || string_01[8]= MetalsGoldFXF
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 576 || string_02[8]= Gold CP for GAIN FIXED spread
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 640 || string_01[9]= MetalsSilverFXF
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 656 || string_02[9]= Silver CP for GAIN FIXED spread
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 720 || string_01[10]= MetalsGoldpro
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 736 || string_02[10]= Gold CP for Forex Pro spread
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 800 || string_01[11]= MetalsSilverpro
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 816 || string_02[11]= Silver CP for Forex Pro spread
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 880 || string_01[12]= ForexPR02
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 896 || string_02[12]= Forex Pro Spreads with tigher spreads
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 960 || string_01[13]= ForexPR03
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 976 || string_02[13]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1040 || string_01[14]= ForexPR04
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1056 || string_02[14]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1120 || string_01[15]= ForexPR05
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1136 || string_02[15]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1200 || string_01[16]= ForexPR06
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1216 || string_02[16]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1280 || string_01[17]= Temp Forex
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1296 || string_02[17]= Temporary location for untradable
symbols
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1360 || string_01[18]= Temp Symbols
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1376 || string_02[18]= Temporary location for FX and CFD
symbols
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1440 || string_01[19]= ForexPR07
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1456 || string_02[19]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1520 || string_01[20]= ForexPR08
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1536 || string_02[20]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1600 || string_01[21]= ForexPR09
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1616 || string_02[21]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1680 || string_01[22]= ForexPR010
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1696 || string_02[22]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1760 || string_01[23]= ForexPR011
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1776 || string_02[23]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1840 || string_01[24]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1856 || string_02[24]=
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 1920 || string_01[25]= CFD - Oil
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 1936 || string_02[25]= Crude Oil
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 2000 || string_01[26]= CFD - Index
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 2016 || string_02[26]= Indexes
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 2080 || string_01[27]= Japan Fixed
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 2096 || string_02[27]= Standard spread for Forex.com Japan
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 2160 || string_01[28]= Japan ForexPro1
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 2176 || string_02[28]= Forex Pro spread for Forex.com Japan
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 2240 || string_01[29]= GoldFXF JP
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 2256 || string_02[29]= Gold CP on FIXED spread for Forex.com
Japan
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 2320 || string_01[30]= SilverFXF JP
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 2336 || string_02[30]= Silver CP on FIXED spread for Forex.com
Japan
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_01= 2400 || string_01[31]= GoldPro JP
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: cur_02= 2416 || string_02[31]= Gold CP on Forex Pro spread for
Forex.com Japan
00:01:35 symgroups_raw_Frame USDJPYFXF, H4: end-----

```

## - 4. &lt; ticks.raw &gt;

## ①コード (ticks\_raw\_Frame.mq4)

```

//+-----+
//| ticks_raw_Frame.mq4 |
//| amenbo |
//| 泉の森の弁財天池 |
//+-----+
#property copyright "amenbo"
#property link "泉の森の弁財天池"
//
int start()
{
    string string_01[], string_02[];
    string SymbolsFileName= "ticks.raw";
    int Offset=0x50;//0d80 バイト
    int SymbolsNumber;

    int hFile = FileOpenHistory(SymbolsFileName, FILE_BIN|FILE_READ);

    Print("start-----");

    FileSeek(hFile, 0, SEEK_SET);
    SymbolsNumber=(FileSize(hFile)/Offset);

    Print("為替ペア数 (SymbolsNumber) は ",SymbolsNumber," です。");

    for(int i=0;i<=(SymbolsNumber-2);i++)
    {
        FileSeek(hFile, (Offset*i), SEEK_SET);

        int cur_01=FileTell(hFile);
        string_01[i+1]= FileReadString(hFile, 12);
        Print("cur_01= ",cur_01," || string_01[" , i+1, "] = ",string_01[i+1]);

    }

    Print("end-----");

    FileClose(hFile);

    PlaySound("alert2.wav");

    return(0);
}
//+-----+

```

## ②結果 (¥experts¥logs¥日付.log)

```

00:02:49 ticks_raw_Frame USDJPYFXF,H4: loaded successfully
00:02:49 ticks_raw_Frame USDJPYFXF,H4: start-----
00:02:49 ticks_raw_Frame USDJPYFXF,H4: 為替ペア数 (SymbolsNumber) は 691 です。
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 0 || string_01[1]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 80 || string_01[2]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 160 || string_01[3]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 240 || string_01[4]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 320 || string_01[5]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 400 || string_01[6]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 480 || string_01[7]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 560 || string_01[8]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 640 || string_01[9]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 720 || string_01[10]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 800 || string_01[11]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 880 || string_01[12]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 960 || string_01[13]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1040 || string_01[14]= USDJPYFXF

```

```

00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1120 || string_01[15]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1200 || string_01[16]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1280 || string_01[17]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1360 || string_01[18]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1440 || string_01[19]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1520 || string_01[20]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1600 || string_01[21]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1680 || string_01[22]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1760 || string_01[23]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1840 || string_01[24]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 1920 || string_01[25]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2000 || string_01[26]= AUDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2080 || string_01[27]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2160 || string_01[28]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2240 || string_01[29]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2320 || string_01[30]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2400 || string_01[31]= AUDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2480 || string_01[32]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2560 || string_01[33]= AUDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2640 || string_01[34]= AUDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2720 || string_01[35]= AUDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2800 || string_01[36]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2880 || string_01[37]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 2960 || string_01[38]= AUDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 3040 || string_01[39]= AUDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 3120 || string_01[40]= GBPJPYFXF
. . . . . (中略 ; あまりに多い) . . . . .
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 54400 || string_01[681]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 54480 || string_01[682]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 54560 || string_01[683]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 54640 || string_01[684]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 54720 || string_01[685]= GBPJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 54800 || string_01[686]= AUDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 54880 || string_01[687]= EURJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 54960 || string_01[688]= EURJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 55040 || string_01[689]= USDJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: cur_01= 55120 || string_01[690]= EURJPYFXF
00:02:49 ticks_raw_Frame USDJPYFXF,H4: end-----

```

## C. 2ファイルの中身をプリントしてみる・・全部は判らない部分が多いけど

## - 1. &lt; symbols.sel &gt; ・・パラメータの入力ウィンドウで為替ペアを指定可

①コード (symbols\_sel\_content\_of\_pair.mq4)

```

//-----+-----
//|                                     symbols_sel_content_of_pair.mq4 |
//|                                     amenbo |
//|                                     泉の森の弁財天池 |
//-----+-----
#property copyright "amenbo"
#property link      "泉の森の弁財天池"

#property show_inputs
//
extern string Symbol_name="USDJPYFXF";//デフォルトの為替ペア
//
int sector=0x80;//=128  1為替ペアのレコード・サイズ
int version_offset=4;//4バイト   ファイルの先頭に4バイトのバージョン情報あり
//
int start()
{
    //
    int hFile = FileOpenHistory("symbols.sel", FILE_BIN|FILE_READ);
    int symbols_number=(FileSize(hFile)-4)/sector;//いくつの為替ペア情報があるか
    //
    FileSeek(hFile,0, SEEK_SET);
    string version=FileReadInteger(hFile, LONG_VALUE);
    Print("Version NO= ",version);
    //
    FileSeek(hFile,version_offset, SEEK_SET);
    //
    for(int i=0;i<symbols_number;i++)
    {

        int offset=FileTell(hFile);
        string short_name=FileReadString(hFile, 12);

        if(short_name==Symbol_name)
        {
            //
            Print("-----START-----");
            //
            Print("ShortName= ", short_name);
            //
            FileSeek(hFile, (offset+0x0c), SEEK_SET);
            int digits_=FileReadInteger(hFile, LONG_VALUE);
            Print("Digits= ", digits_);
            //
            FileSeek(hFile, (offset+0x20), SEEK_SET);
            double point_=FileReadDouble(hFile, DOUBLE_VALUE);
            Print("Point= ", point_);
            //
            FileSeek(hFile, (offset+0x38), SEEK_SET);
            datetime time_current=FileReadInteger(hFile, LONG_VALUE);
            Print("TimeCurrent() in symbols.sel_file= ", time_current);
            //
            int year_=TimeYear(time_current);
            int month_=TimeMonth(time_current);
            int day_=TimeDay(time_current);
            int hour_=TimeHour(time_current);
            int minute_=TimeMinute(time_current);
            int second_=TimeSeconds(time_current);
            Print("Time_in_symbols.sel_file:",year_,"年",month_,"月",day_,"日",hour_,"時",minute_,"分",second_,"秒");
            //-----
            Print("current server time:",Year(),"年",Month(),"月",Day(),"日",Hour(),"時",Minute(),"分",Seconds(),"秒");
            //
            int year_p=TimeYear(TimeLocal());
            int month_p=TimeMonth(TimeLocal());
        }
    }
}

```

```

int day_p=TimeDay(TimeLocal());
int hour_p=TimeHour(TimeLocal());
int minute_p=TimeMinute(TimeLocal());
int second_p=TimeSeconds(TimeLocal());
Print("PC time:", year_p, "年", month_p, "月", day_p, "日", hour_p, "時", minute_p, "分", second_p, "秒");
//-----
FileSeek(hFile, (offset+0x40), SEEK_SET);
double bid_1=FileReadDouble(hFile, DOUBLE_VALUE);
double ask_1=FileReadDouble(hFile, DOUBLE_VALUE);
Print("Bid_1= ", bid_1, " || ask_1= ", ask_1);
//
FileSeek(hFile, (offset+0x50), SEEK_SET);
double high_0=FileReadDouble(hFile, DOUBLE_VALUE);
double low_0=FileReadDouble(hFile, DOUBLE_VALUE);
Print("High[0]= ", high_0, " || Low[0]= ", low_0);
//
FileSeek(hFile, (offset+0x70), SEEK_SET);
double bid_2=FileReadDouble(hFile, DOUBLE_VALUE);
double ask_2=FileReadDouble(hFile, DOUBLE_VALUE);
Print("Bid_2= ", bid_2, " || ask_2= ", ask_2);
//
//
Print("----Gap List----");
FileSeek(hFile, (offset+0x10), SEEK_SET);
int gap1=FileReadInteger(hFile, LONG_VALUE);
int gap2=FileReadInteger(hFile, LONG_VALUE);
int gap3=FileReadInteger(hFile, LONG_VALUE);
int gap4=FileReadInteger(hFile, LONG_VALUE);
Print("Gap1~Gap4= ", gap1, ";", gap2, ";", gap3, ";", gap4);
//
FileSeek(hFile, (offset+0x28), SEEK_SET);
int gap5=FileReadInteger(hFile, LONG_VALUE);
int gap6=FileReadInteger(hFile, LONG_VALUE);
int gap7=FileReadInteger(hFile, LONG_VALUE);
int gap8=FileReadInteger(hFile, LONG_VALUE);
Print("Gap5~Gap8= ", gap5, ";", gap6, ";", gap7, ";", gap8);
//
FileSeek(hFile, (offset+0x3c), SEEK_SET);
int gap9=FileReadInteger(hFile, LONG_VALUE);
Print("Gap6= ", gap9);
//
FileSeek(hFile, (offset+0x60), SEEK_SET);
double gap10=FileReadDouble(hFile, DOUBLE_VALUE);
double gap11=FileReadDouble(hFile, DOUBLE_VALUE);
Print("Gap7, Gap8= ", gap10, ";", gap11);
//
Print("-----END----");
//
break;
}
FileSeek(hFile, (sector-12), SEEK_CUR);
}

FileClose(hFile);

PlaySound("alert2.wav");
return(0);
}
//-----

```

②結果 (¥experts¥logs¥日付.log)

```

00:07:15 symbols_sel_content_of_pair USDJPYFXF, H4: loaded successfully
00:07:23 symbols_sel_content_of_pair USDJPYFXF, H4 inputs: Symbol_name="USDJPYFXF";
00:07:23 symbols_sel_content_of_pair USDJPYFXF, H4: Version NO= 400
00:07:23 symbols_sel_content_of_pair USDJPYFXF, H4: -----START-----
00:07:23 symbols_sel_content_of_pair USDJPYFXF, H4: ShortName= USDJPYFXF

```

```

00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Digits= 2
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Point= 0.01
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: TimeCurrent() in symbols_sel_file=
1367000561
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Time_in_symbols_sel_file:2013年4月26日
18時22分41秒
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: current server time:2013年4月27日0時7
分16秒
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: PC time:2013年4月27日0時7分23秒
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Bid_1= 98.73 || ask_1= 98.75
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: High[0]= 99.49 || Low[0]= 98.22
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Bid_2= 98.73 || ask_2= 98.75
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: ----Gap List----
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Gap1~Gap4= 263;1;4;0
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Gap5~Gap8= 0;0;1;256
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Gap6= 0
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: Gap7,Gap8= 0;0
00:07:23 symbols_sel_content_of_pair USDJPYFXX,H4: -----END-----

```

## － 2. < symbols.raw > ・ ・パラメータの入力ウインドウで為替ペアを指定可

### ①コード (symbols\_raw\_content\_of\_pair.mq4)

```

//-----+-----
//|                                     symbols_raw_content_of_pair.mq4 |
//|                                     amenbo |
//|                                     泉の森の弃財天池 |
//+-----+-----
#property copyright "amenbo"
#property link      "泉の森の弃財天池"

#property show_inputs
//
extern string Symbol_name="USDJPYFXX";//デフォルトの為替ペア
int record=0x790;//=1936 1為替ペアのレコード・サイズ

int start()
{
    int hFile = FileOpenHistory("symbols.raw", FILE_BIN|FILE_READ);
    int symbols_number = FileSize(hFile)/record;

    FileSeek(hFile, 0, SEEK_SET);

    for(int i=0;i<symbols_number;i++)
    {
        int offset=FileTell(hFile);
        string short_name=FileReadString(hFile, 12);

        if(short_name==Symbol_name)
        {
            //
            Print("-----START-----");
            //
            Print("ShortName= ", short_name);
            //
            string Description=FileReadString(hFile, 64);
            Print("Description= ", Description);
            //
            FileSeek(hFile, (offset+0x0690), SEEK_SET);
            double swap_long_value=FileReadDouble(hFile, DOUBLE_VALUE);
            Print("swap_long_value= ", swap_long_value);
            //
            FileSeek(hFile, (offset+0x0698), SEEK_SET);
            double swap_short_value=FileReadDouble(hFile, DOUBLE_VALUE);
            Print("swap_short_value= ", swap_short_value);
        }
    }
}

```

```

//
FileSeek(hFile, (offset+0x06a8), SEEK_SET);
double contract_size=FileReadDouble(hFile, DOUBLE_VALUE);
Print("contract_size= ", contract_size);
//
FileSeek(hFile, (offset+0x06c0), SEEK_SET);
int stops_level=FileReadInteger(hFile, CHAR_VALUE);
Print("stops_level= ", stops_level);
//
Print("-----END----");
//
break;
}

FileSeek(hFile, (record-12), SEEK_CUR);

}

FileClose(hFile);
PlaySound("alert2.wav");

return(0);
}

```

②結果 (¥experts¥logs¥日付.log)

```

00:09:05 symbols_raw_content_of_pair USDJPYFXF, H4: loaded successfully
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4 inputs: Symbol_name="USDJPYFXF";
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4: -----START-----
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4: ShortName= USDJPYFXF
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4: Description= US Dollar vs Japanese Yen
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4: swap_long_value= 0
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4: swap_short_value= -0.4
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4: contract_size= 100000
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4: stops_level= 1
00:09:15 symbols_raw_content_of_pair USDJPYFXF, H4: -----END----

```

## 4. 補足

### (1) MT 4 をオフラインにする方法

・「過激な方法」を除外すると、簡単にオフラインにするには通常下記の方法があります。

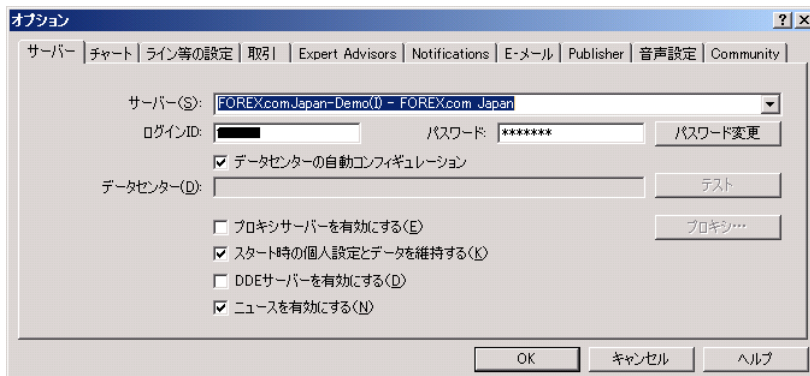
まず、[ツール] - [オプション] で「オプション」ウインドウを表示し、  
下記の何れかを実行する。

- A. 「プロキシサーバーを有効にする」にチェック
- B. サーバー ; 空欄にする
- C. ログイン ; パスワードを空欄にする

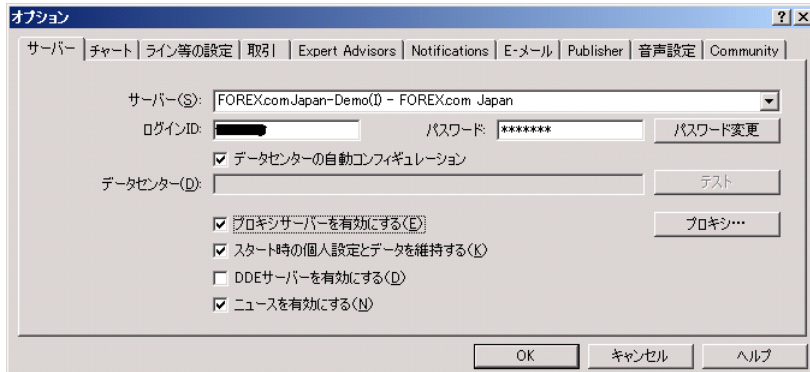
※本稿では使いやすいので「A.」の方法を解説します。

(良く知られた方法なので、ご存知でしたら、ご容赦)

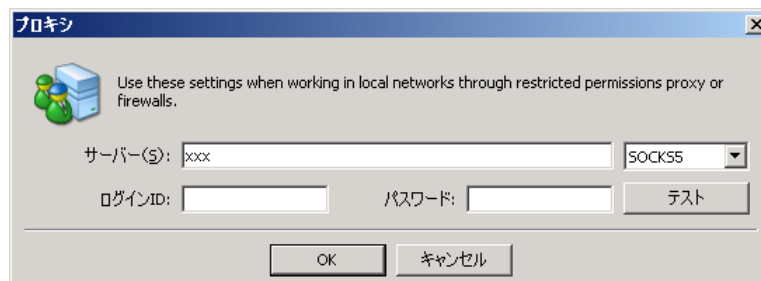
- ① [ツール] - [オプション] で「オプション」ウインドウを表示



- ② [プロキシサーバーを有効にする] にチェック



- ③ 「サーバー」に在り得ない適当な文字（本例では「xxx」）を入れる。

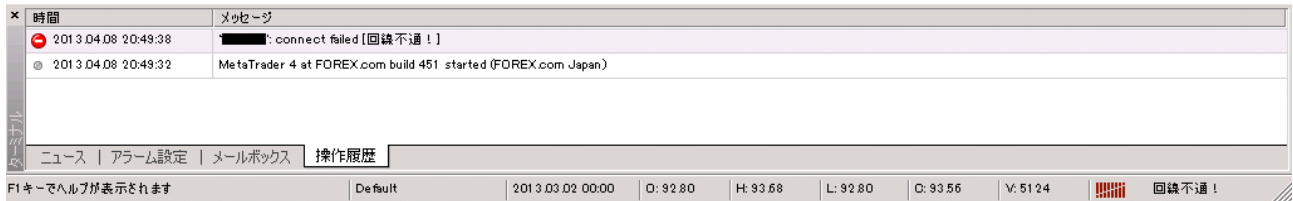


- ④ MT4 をシャットダウンする

- ⑤ MT4 を再起動する

・ ・ 確認すると、





※ [回線不通] となります。

## (2) ライブ・トレード用の「contract specification (契約仕様)」では、spread が変!

・もしかすると、「変動スプレッド」の現状では、以下の結果は当然なのかもしれませんが!

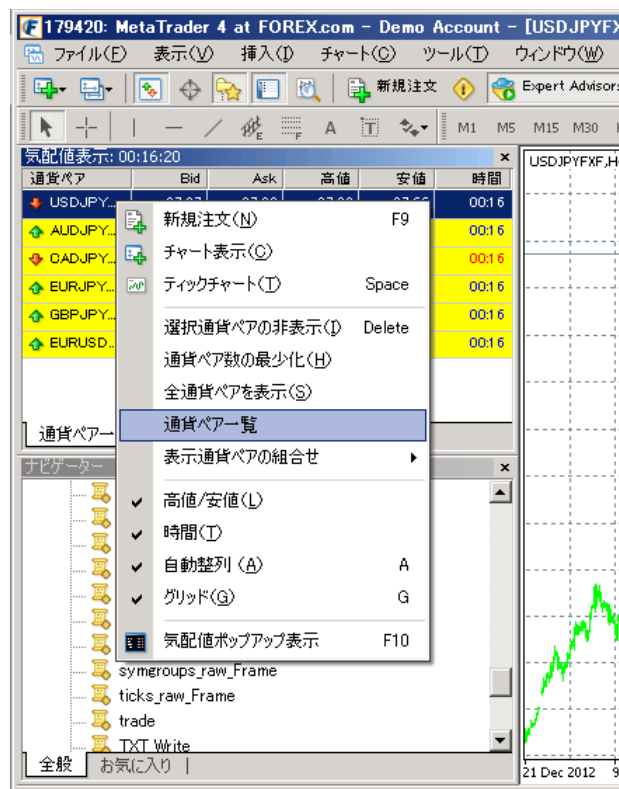
※以下に、ほぼ同時刻での

- 1. ライブ・トレード用の「contract specification (契約仕様)」
- 2. [注文の発注] ウィンドウ
- 3. バックテスト用の「contract specification (契約仕様)」

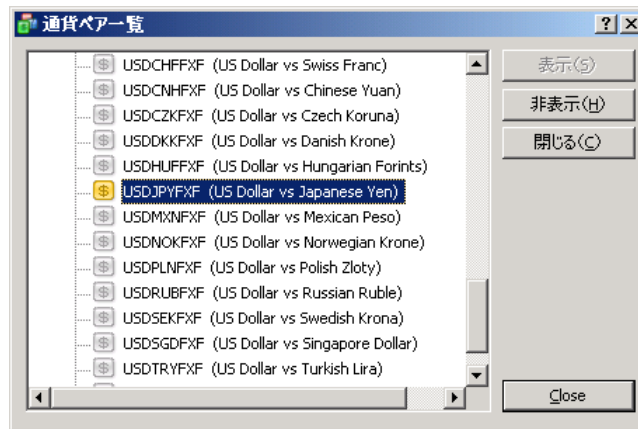
を比較します。

### - 1. ライブ・トレード用の「contract specification (契約仕様)」を表示する

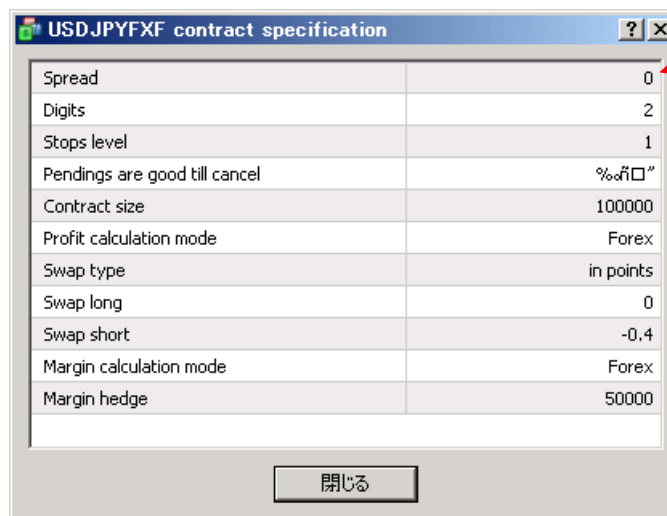
①MarketWatch ウィンドウ上の「通貨ペア」を選択し、右クリック



② [通貨ペア一覧] を選択



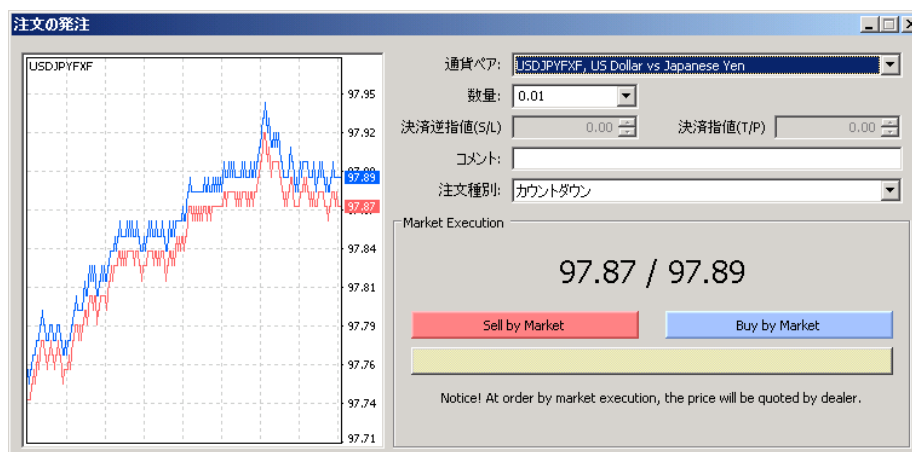
③通貨ペアを選択し、[閉じる] を選択



※ 「Spread」 が「0」だ！！・・実は、他の通貨ペアでも「0」

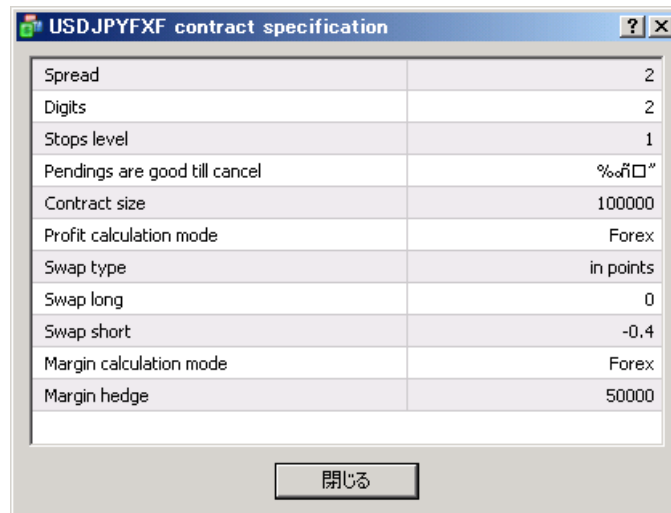
ー 2. [注文の発注] ウィンドウ

・チャート上で右クリックし、一番上段の [注文の発注] を選択



※ 「Spread」 は「2」だ！

### － 3. バックテスト用の「contract specification (契約仕様)」



#### <結論>

- ライブ・トレード用の「contract specification (契約仕様)」で、スプレッド (Spread) が「0」であるのは、「バグ」ではなくて、「変動スプレッド」に対応した結果と思われる。
- 変動スプレッド (Spread) の現在値は、[注文の発注] ウィンドウで確認する。
- バックテスト用の「contract specification (契約仕様)」は、MT4 では固定値なので、スプレッド (Spread) が「2」と設定されている。

#### (3) 次回 (その2) の予定

- ・「¥history¥」ローカ名 (Live、Demo) 中の4システム・ファイル、特に「symbols.sel」と「バックテスト」との係りを中心に調べ、報告する予定です。(現在調査・実験中です)

以 上