

Highlights of version 5.50

- **New Analysis window** introduced in version 5.50 brings the following improvements over old Automatic Analysis
 - **multi-threaded operation = speed** - new Analysis window uses all available CPUs/cores to execute formulas in many threads in parallel providing significant speed ups. For example on 4 core Intel i7 that can run up to 8 threads, it can run up to 8 times faster than old Analysis window. Exact speed up depends on complexity of the formula (the more complex it is, the more speedup is possible), amount of data processed (RAM access may be not as fast as CPU thus limiting possible speed gains).
 - **non-blocking operation** - you can now view, scroll and sort results of analysis while they are still generated, also as user interface thread is not used for processing for most part, charts and other GUI-driven program parts are way more responsive than with old automatic analysis
 - **multiple instances** - you can run more than one instance of New Analysis at a time, so you can run many scans/backtest/explorations/optimizations in parallel without waiting for one to complete
 - **slicker user interface** - New Analysis window can act as tabbed document, can be floated, buttons can be re-arranged for better workflow. There is way more space for the result list, extra information about execution is provided on the new "Info" tab. Also walk-forward results are now displayed within New Analysis window for less clutter.
- Mini **High-Low rank chart** in Real Time quote window
- User-definable **mini bar charts in Explorations** (see AddColumn function)
- **Add Rank Column** feature - right-click Analysis result list and choose "Add Rank column" - it adds a column with ordinal rankings based on current sort or just row number column when list is not sorted
- IRA account backtesting via SettlementDelay feature (see SetOption function)
- **Range bars** algorithm improved significantly
- **new AFL functions:** ThreadSleep, StaticVarCompareExchange
- **updated AFL functions** with new functionality: AddColumn, SetOption, GetOption, CategoryGetSymbols, PopupWindow, GetFnData, ClipboardSet
- updated OLE interface to support new Analysis window
- updated custom backtester interface to support access to local, per-analysis EquityArray property
- updated UI in many places
- Owner-draw list views for 10x speed improvement when displaying millions of rows
- many other improvements (see Release Notes for details)

CHANGES FOR VERSION 5.50.5 final release (as compared to 5.50.4)

1. Newly created chart template did not save current tab when it was not switched before saving. Fixed.
2. Notepad window is refreshed when it is brought back from invisible tab
3. AFL Code Wizard: sometimes 64-bit version of AFL wizard could hang on clicking "AND/OR" rule. Fixed (AFL Wizard 1.0.2)
4. Charts: Invisible Symbol-linked charts may not be properly refreshed when changing symbol (happens only when not using streaming RT source). Fixed.
5. OLE: AnalysisDoc.Run now supports mode =3 (individual backtest)
6. AFL: Equity()/Status()/GetOption used on chart referred to OLD AA settings instead of last run analysis (whenever it is old AA or new). Now uses last run analysis settings
7. AFL: SetFormulaName did not work in New Analysis. Fixed

CHANGES FOR VERSION 5.50.4 (as compared to 5.50.3)

1. New Analysis: "Show arrows for actual trades" sometimes used date range from old AA. Fixed
2. New Analysis: Individual backtest – custom metrics columns would not appear if very first symbol under test does not generate ANY trades. Fixed.
3. UI: when you float a maximized MDI window it becomes little smaller so it does not overlap MDI tabs.
4. Print/Print Preview: circle/arc/rectangle objects outside current view are properly clipped on printouts
5. New Analysis: When user cancelled optimization in the middle and ran backtest afterwards then this single follow up backtest could be incorrect and required another run to produce correct result again. Fixed.
6. When timeshift != 0 then duplicate range markers could appear at day boundary. Fixed.
7. Small rounding issue in ray drawing fixed
8. When multiple chart panes were using SetForeign on rare occasions foreign call could fail (return other symbol data) Fixed.

CHANGES FOR VERSION 5.50.3 (as compared to 5.50.2)

- New Analysis: proper symbol list refresh is triggered when composites are created or user calls SetOption("RefreshWhenCompleted, True);
- When settlement delay was set to zero, then interest rate setting was ignored (5.46-5.50.2). Fixed.
- New Analysis: Individual backtest: setting different report mode took place only after second run / second symbol. Fixed.
- Formulas: styleNoLabel is now default for all "Band" formulas (Bollinger/Keltner/Percent)
- Formulas: faster version of Portfolio Equity formula
- Formulas: Ichimoku cloud added
- TC2K/TC2KFunds plugins updated to check Wow6432Node registry key on 64-bit installations (should avoid TC2k detection issue on 64-bit Windows)
- PlotVAPOverlay did not draw the very last level line. Fixed. Also done some cosmetic tweaks.
- New Analysis: Walk forward tab table did not display custom metrics. Fixed.

- Ticker bar prevents from adding empty " symbol names
- Blank chart can be closed using flyout minitoolbar now
- When database is empty or chart does not have properly selected symbol the information note is displayed in the chart window (instead of blank chart)
- Fixed crash @7C8097FF when trying to drag drop indicator on chart when database is empty

CHANGES FOR VERSION 5.50.2 (as compared to 5.50.1)

- New Analysis: If current formula is open in the formula editor it is auto-saved prior to running any Analysis (as old AA did)
- New Analysis: on Backtest/Optimization current formula is saved in @LastBacktestFormula registry key so it can be referenced by Individual Equity formula
- New Analysis and Old AA: HoldMinBars > 0 can now be used together with AllowSameBarExit=True Settings: 'Stops' tab: N Bar stop has priority check box state was not saved. Fixed
- New Analysis: on attempt to load analysis project file that contains formula path that does not exist or is not writable AmiBroker will create the formula in the "Formulas\Imported" folder instead of not creating it at all
- New Analysis: WinXP-only - RMB click column descend/ascend menus were not working due to Windows XP bug. Fixed.
- New Analysis: when formula is picked from the folder that is inside current working directory, then relative path is used and stored in project instead of absolute path previously used
- New Analysis: SeparateLongShortRank required proper MaxLongPos and MaxShortPos definitions in order to work. Now if they are missing defaults are applied as in old AA.
- Plugins: IQFeed: added "You can't close plugin window" on attempt to close working IQFeed window.
- UI: Chart tree, Layout tree in "Explorer" theme
- UI: Category list now features new icons and decreased indentation
- UI: Owner-draw listviews now fully implement ellipsis item truncation mechanism, but using ExtTextOut only which is 6 times faster than DrawText that Windows uses
- UI: Floating non-chart windows (account/web browser/analysis) are not closed when switching layouts.
- Plugins: IQFeed plugin uses new faster IQFeed API, downloads less and uses way more efficient method to backfill 1/5/15-second base time intervals

CHANGES FOR VERSION 5.50.1 (as compared to 5.50.0)

- New Analysis: Individual Backtest implemented. Note that it runs with approx same speed as old AA as Individual Backtest is single threaded for reasons explained in the User's manual http://www.amibroker.com/guide/h_multithreading.html
- New Analysis: toolbar icons for parameters and settings changed to avoid confusion with the icon for preferences
- UI: Symbol List uses Windows 7/Vista style
- New Analysis: Toolbar customization is now saved
- MRU (most recently used) file menu moved to submenu and MRU list increased to 10 items

- Charts: fixed small alignment problem on Y axis when using fixed (nonproportional) font
- New Analysis: backtester object now has EquityArray property that returns entire equity array (not just current value).
Please note that values are filled during backtest and only after backtest is complete, all values are valid. If you call it in the middle, it will contain only "upto given point" data. Avoid abusing this function and it is costly in terms of RAM/CPU.e).
You may use
bo.EquityArray instead of Foreign("~~~Equity", "C")
in custom backtester code.
It accesses local copy of equity in New Analysis (unlike Foreign that accesses global symbol)
- New Analysis: AddToComposite with atcDeleteValues | atcEnableInPortfolio did not delete previous values. Fixed
- New Analysis: "Interval" menu item in the auto-repeat group renamed to "AR interval" because some users interpreted it wrong (as periodicity)
- New Analysis: OLE interface implemented
Description in the <http://www.amibroker.com/guide/objects.html>

Example

The following JScript example

- a) opens analysis project from C:\Analysis1.apx file
- b) starts backtest (asynchronously)
- c) waits for completion
- d) exports results
- e) closes analysis document

```
AB = new ActiveXObject( "Broker.Application" );

try
{
    NewA = AB.AnalysisDocs.Open( "c:\\analysis1.apx" ); // opens given
analysis project file

    if ( NewA )
    {
        NewA.Run( 2 ); // starts analysis (asynchronous - returns
immediatelly
        // (0-scan, 1- exploration, 2- portfolio backtest, 4-
optimization, 6-walk forward)

        while ( NewA.IsBusy )
            WScript.Sleep( 500 ); // wait for completion

        NewA.Export( "test.html" ); // exporing results

        WScript.echo( "ended" );

        NewA.Close(); // closing analysis window
    }
}
catch ( exc )
{
    WScript.echo( "Exception: " + exc.message );
}
```

- Charts: more tweaks for Y axis algorithm for readability. Now in Tools->Preferences, "Charting" the user may select different grid stepping: Instead of default 1, 2, 5, 10 grid stepping, user may select to use 2.5 step instead of 2 (Use 2.5 step in Y axis)
- GetFnData() supports new field "Alias" - return Alias field without need to use OLE
- GetOption supports new fields: ApplyTo and Filter*, eliminates the need to use OLE for that purpose
New fields:
- "ApplyTo" - returns Analysis "Apply To" setting 0 - all symbols, 1 - current symbol, 2 - filter
- "FilterIncludeIndex", "FilterIncludeFavorite", "FilterIncludeMarket", "FilterIncludeGroup", "FilterIncludeSector", "FilterIncludeIndustry", "FilterIncludeWatchlist" - return "Include" filter settings -1 - means NOT selected (not included), >= 0 index of included category
- "FilterExcludeIndex", "FilterExcludeFavorite", "FilterExcludeMarket", "FilterExcludeGroup", "FilterExcludeSector", "FilterExcludeIndustry", "FilterExcludeWatchlist" - return "Exclude" filter settings -1 - means NOT selected (not excluded), >= 0 index of excluded category
- Fix for FPU inaccuracies in range bars implemented

CHANGES FOR VERSION 5.50.0 (as compared to 5.49.2)

- New Analysis: "Save" creates now a Analysis project (.apx) file that includes all settings and formula needed in single file. The file itself is in human readable XML format. When such file is loaded on somebody's else computer it will recreate original formula if one is not present. It will warn if there is a name conflict and let you decide to keep existing formula or overwrite.
- UI: File open dialog now has "All supported files" option in the "Files of type"
- UI: PopupWindow is now resizable (see size gripper in the lower-right corner)
- AFL: PopupWindow function has additional parameters to control width/height and flag that controls whenever popup window captures input focus
PopupWindow("bodytext", "captiontext", timeout = 5, left = -1, top = -1, width = -1, height = -1, captureFocus = True);

width = width in pixels, -1 - use default width

height = height in pixels, -1 - use default height

captureFocus - decides whenever popup window captures input focus or not

Example: popup window that does not change input focus:

PopupWindow("test", "caption", 30, -1, -1, -1, -1, False);

- New Analysis: ApplyStops applied in non-regular modes are passed to 2nd phase properly now.
- New Analysis: "Show current trade arrows" fixed.
- New Analysis: View->Filtering menu options were not available when New Analysis was active. Fixed.
- New Analysis: Standard Edition has a limit of max 2 threads per Analysis window instance.
- UI: Log window clicking on error displayed AFL Editor and marked Line+1, Col+1 char position instead of proper place. Fixed

- New Analysis: Errors are now displayed in a modern message bar instead of dialog box
- UI: holding down Ctrl and Q keys simultaneously while AmiBroker is starting resets all menus/command bars to defaults
- New Analysis: scroll bar properly works when "Walk forward" tab is active
- New Analysis: Walk forward tab content can be now copied using Edit->Copy / Ctrl+C
- New Analysis: when "wait for backfill" was turned on and external data source was very slow to repond (like IQFeed during backfill) subsequent calls to Status('stocknum') could give non-unique numbers. Fixed.
- New Analysis: Copy to clipboard includes headers now
- New Analysis: Column customization/state persistence implemented
- Charts: New Y axis grid algorithm for clearer display on condensed charts
- AFL: CategoryGetSymbols supports new categoryAll (all symbols in the database) and new Mode parameter
CategoryGetSymbols(category, number, mode = 0);

Mode parameter decides what field is retrived:

0 (default value) - ticker symbol

1 - full name

Example: to get all symbols existing in the database simply call

CategoryGetSymbols(categoryAll, 0);

to get full names of all symbols use:

CategoryGetSymbols(categoryAll, 0, 1);

- AFL: Added Warning 503. Using OLE / CreateObject / CreateStaticObject is not multi-threading friendly. See Users' Guide ""Efficient use of multithreading"" for more details.
- AFL: Added ThreadSleep(milliseconds) function. It suspends current thread for specified number of milliseconds (maximum is 100 ms). Works only from NON-UI threads. When called from UI thread the function does NOTHING and returns immediatelly. Please do NOT abuse this function. Using it may negatively impact performance.
- AFL: Added StaticVarCompareExchange function. Provides atomic interlocked compare/exchange functionality

Parameters

" varname "

Specifies the name of the destination static variable. Static variable if exists must be scalar numeric type. If static variable is not initialized, the function assumes that it has value of zero.

exchange

Specifies the exchange value. Scalar numeric.

Comperand

Specifies the value to compare to the destination static variable. Scalar numeric.

Return Values

The return value is the initial value of the destination static variable. If variable did not exist, it returns zero.

The StaticVarCompareExchange function performs an atomic comparison of the "varname" static variable value with the Comperand value. If the static variable value is equal to the Comperand value, the Exchange value is stored in the static variable. Otherwise, no operation is performed.

The function StaticVarCompareExchange provides a simple mechanism for synchronizing access to static variables that are shared by multiple threads. The following examples show how to implement semaphore and critical section in AFL using StaticVarCompareExchange function:

```
// EXAMPLE 1 : Simple semaphore (no waiting)
if( StaticVarCompareExchange( "semaphore", 1, 0 ) == 0 ) // obtain
semaphore
{
    // protected section here
    // Here you have exclusive access (no other threads that check for
semaphore will enter simultaneously)
    ////////////////////////////////
    StaticVarSet("semaphore", 0 ); // reset semaphore
}
else
{
    _TRACE("Can not obtain semaphore");
}

////////////////////////////////
// EXAMPLE 2 HOW TO IMPLEMENT CRITICAL SECTION IN AFL
////////////////////////////////

function _TryEnterCS( secname )
{
    global _cursec;
    _cursec= "";

    // try obtaining semaphore for 1000 ms
    for( i = 0; i < 1000; i++ )
        if( StaticVarCompareExchange( secname, 1, 0 ) == 0 )
        {
            _cursec = secname;
            break;
        }
        else ThreadSleep( 1 ); //sleep one millisecond

    return _cursec != "";
}

// call it ONLY when _TryEnterCS returned TRUE !
function _LeaveCS()
{
    global _cursec;
    if( _cursec != "" )
    {
        StaticVarSet( _cursec, 0 );
        _cursec = "";
    }
}
```

```

function TimeConsumingWork()
{
    // WARNING: the Percentile is CPU hungry as it involves lots of
    // sorting, the loop below may take > 1 second to complete
    for( i = 0; i < 10; i++ ) Percentile( C, 100, 10 );
}

//_TRACE("Without CS Begin " + GetChartID() );
//TimeConsumingWork(); // some time consuming calculation
//_TRACE("Without CS End" + GetChartID() );

// Example usage (critical section)
if( _TryEnterCS( "mysemaphore" ) )
{
    // you are inside critical section now
    _TRACE("Begin CS " + GetChartID() );
    TimeConsumingWork(); // some time consuming calculation
    _TRACE("End CS " + GetChartID() );
    _LeaveCS();
}
else
{
    _TRACE("Unable to enter CS");
}

```

CHANGES FOR VERSION 5.49.2 (as compared to 5.49.1)

- New Analysis: added Optimize->3D Optimization chart menu
- New Analysis: buttons for Portfolio Equity/Individual Equity charts added
- New Analysis: Info Bar added (that duplicates some of the info that normally appears on the Info Tab). It can be turned on using Settings->Info Bar
- New Analysis: Report generation is now OFF by default when running optimization/walk forward as in old AA. This gives about 0.9 sec gain per step.
- New Analysis: Single-symbol optimization made faster by caching already compressed quote array on subsequent steps
- New Analysis: the check for Status("StockNum") was sensitive to blanks between parenthesis, function name and string parameter. Now it is made insensitive to blanks.
- New Analysis: when "Range" was "n last bars" or "n last days", Status("rangefromdate") gave date of preceding bar. Fixed.
- UI: "Performance" tooltip title was not always updated on time. Fixed.
- UI: "Wait for plugin" window displays progress bar with "marquee" style instead of circle of dots

CHANGES FOR VERSION 5.49.1 (as compared to 5.49.0)

- New Analysis: Testing on unaligned data sometimes took longer than expected because of the work necessary to bring signals in order. Now a smart method avoids most of the memory transfers and can give 10x speedup on backtesting very long unaligned intraday data.
- New Analysis: "Backtest started", "Exploration started" text in the info tab appeared twice per run (instead of once). Fixed
- New Analysis: Backtest report did not include formula code. Fixed.
- New Analysis: Fixed crash that could occur when running analysis with Range set to "N recent bars" and N was exceeding number of quotes.
- New Analysis: Float/Normal/Close popup menu RMB menu was active everywhere in floating window instead of only caption area. Fixed.

- New Analysis: Implement single-thread start when Status("StockNum") is detected in the code. Other threads are started when first symbol processing is completed.
- New Analysis: Send to Analysis did not update formula combo box properly in certain scenarios: when formula X was selected, then formula Y, then formula X again. Fixed.
- Old AA: status bar displays true timing for fair comparisons between old AA and new Analysis. Watching progress bar numbers was misleading as it closed before all processing was truly complete, so it looked like old AA completed sooner than it really did
- UI: menu item for Old Automatic Analysis restored as apparently some users got lost. Also attempt to run old Automatic analysis displays an information about new Analysis window

CHANGES FOR VERSION 5.49.0 (as compared to 5.48.0)

- New Analysis: when floating pane is made normal from/to dates were reset to today. Fixed.
- New Analysis: when Apply To: Filter is defined so NO symbol matches the criteria, appropriate message is displayed when trying to run Analysis and action is terminated instead of displaying progress bar forever
- New Analysis: when analysis window is "floating" double click/show arrows/sync affects active MDI chart just like old AA did, as opposed to last one (when analysis is not floating)
- New Analysis: Optimization parameter setup stage uses now not more than 100 bars. This reduces setup time.
- New Analysis: implemented import / export, File menu now has Import HTML and Export HTML/CSV commands
- New Analysis: Implemented "Wait for backfill" option. This slows down analysis sequence so AmiBroker waits for external RT data source to deliver backfill data. When "wait for backfill" option is turned on, AmiBroker checks if data for given symbol are complete, if not, it sends request to external source to backfill the data and waits 1 second. After one second, it checks again, if data are complete, if not, it waits another 1 second, and so on. Once data are complete given symbol is used for formula execution.
- New Analysis: Custom backtester uses "last backtest settings" when calling GetStatus() instead of old AA settings
- New Analysis: auto-repeat interval edit field did not allow to enter values if moved to main toolbar via customization. Fixed.
- Formula Editor Analysis toolbar button now operates on New Analysis window
- Default for timestamping of intraday compression changed to "Show START time of interval"
- Charts: AFL error message color is now definable in Tools->Preferences, "Color" tab
- Charts: Tooltips automatically display also OHL prices (in addition to close) when Plot style is set to candlestick, bar or when PlotOHLC was used
- UI: Adjusted item Y offset for owner drawn listviews for consistent look with old layout of system listviews in Windows XP and earlier OSes
- UI: Account, Analysis and Web research windows don't use client edge/border anymore - gives slicker / less crowded look
- UI: when focus is switched to floating document window, the application menu is switched to proper context

- UI: When File->Close was used on floating pane, document was closed but docking pane was not. Fixed.
- UI: The toolbar button and menu option to launch old Automatic Analysis window are now hidden (can be brought back by using Tools->Customize), old AA can also be launched from new Analysis via Analysis->Old Analysis menu
- UI: New menu options and toolbar button to launch new analysis window
- UI: Moved all menu items controlling visibility of docking windows to "Window" menu
- UI: floating state of windows can now be turned on/off using right-click menu over tab / floating window caption
- UI: File->Save (when currently selected window is a chart) command now saves Chart template. To save database use "Save database" instead.
- UI: Defaults for data window are changed, so extra OHL display is turned off by default as currently PlotOHLC displays all 4 prices in the tooltip and data window already
- UI: Data window scroll bars are hidden and do not flicker when updating
- UI: "Save All" toolbar button removed, "Save database" button added to replace it
- Web Research: turned on "silent" mode - should prevent error message boxes displayed by IE engine.
- QuoteEngine: Range bars algorithm improved significantly
- Quote array uses 20480 bytes (512 quotes) allocation chunk size (exactly 5x4KB page) for better alignment with Windows virtual memory pages

CHANGES FOR VERSION 5.48.0 (as compared to 5.47.0)

- New Analysis: custom metrics columns were invisible in backtest/optimizations. Fixed. (#91031)
- New Analysis: backtest did not display proper column names when Report mode was different than trade list. Fixed. Also added support for changing report mode via SetOption().
- New Analysis: Auto-repeat Scan/Explore implemented (aka. "Run every" in old AA) (use drop down menu next to "Settings" button)
- New Analysis: Showing arrows by double click implemented
- New Analysis: Sync chart on select option implemented (use drop down menu next to "Settings" button)
- Fixed window activation issue after RMB click in chart window when non-chart floating windows are active
- New Analysis: Cancelling walk-forward did not restore date range and settings and didn't reset internal step counter. Fixed.
- New Analysis: Formula path disappeared when window was made floating. Fixed.
- New Analysis: in 5.47 non-exhaustive optimizers sometimes stopped after first step in walk-forward. Fixed.
- Charting: scroll bar zoom function remains active even at extreme zoom in levels
- New Analysis: Fixed problem with opt steps warning dialog appearing sometimes in the out-of-sample tests in walk forward sequence
- New Analysis: Signal heaps are separated now, allowing to run concurrent backtest/optimizations in many analysis windows
- New Analysis: removed unnecessary refreshes of symbol list during optimizations that caused slowdowns when user had large database
- New Analysis: The size of control fields is properly adjusted with regards to DPI

- New Analysis: Status("StockNum") was always zero in 5.47 and earlier. Now it is set properly.
Caveat: be careful when you run codes that use Status("StockNum")==0 for special processing in multithreading mode.
Other threads with "StockNum" = 1, 2, 3,... will run simultaneously! If you store static variables in step 0 these variables may be empty if you access them from other parallel running threads.
- Fixed Error 53 message that was giving an error when proper \" (quotation mark espace sequence) was used

CHANGES FOR VERSION 5.47.0 (as compared to 5.46.0)

- New Analysis: last used ApplyTo/Range/From/to dates/filters and other settings are saved when Analysis window is closed
- Fix to "slow response" problems some users reported when switching symbols
- Fibonacci retracement- modifying Z-order affected only control trendline, not all lines as it should. Fixed.
- New Analysis: implemented Walk-Forward testing (use drop arrow next to Optimization button)
- AFL engine: new error 53: incorrect \ espace sequence. Error is issued when user forgets that single slash must be written as \\
- Default title for new analysis window is "AnalysisX" (instead of "Unnamed") where X is consecutive number 1, 2..
- attempt to use Say() on computer without sound card or without driver installed does not result in throwing exception/crash report, but it is rather silently ignored

CHANGES FOR VERSION 5.46.0 (as compared to 5.45.0)

- **New Analysis: Exhaustive multi-threaded optimization implemented**
- **New Analysis: Smart (non-exhaustive) multi-threaded optimization implemented in a way compatible with existing optimizer plugins**

Note however that due to the fact that most non-exhaustive optimization plugins are not re-entrant, AmiBroker prevents from running more than one non-exhaustive multi-threaded optimization at a time. You can however, run many exhaustive optimizations in parallel, as they don't require external plugins.

- Slight modification to the way how compressed bar charts are drawn (so bars with H==L are better visible)

CHANGES FOR VERSION 5.45.0 (as compared to 5.44.1)

- 64-bit version compiled with new version of C runtime (Microsoft security update)
- Fixed date/time column sorting in virtual list views when regional date format was different than yyyy-MM-dd
- AFL: ClipboardSet() returns True (1) when clipboard has been successfully set. Sometimes clipboard is locked by another applications and then write can fail (ClipboardSet() will return zero/False then)
- New Analysis: Changed alignment of Date/Time column to left in backtest result list.
- Most AFL engine setup moved to worker threads. Result - smaller UI thread load in MT charts (better scalability) and much faster (upto 2x) explorations/scans in New Analysis window

- New Analysis: added "Info" tab that provides some summary information about analysis (number of rows, timings, backtest summary) - contents can be copied using Ctrl+C / Edit->Copy

NOTE:

End users should only look at "Completed in... sec" line that shows actual run time of the analysis (scan/exploration/backtest).

The times shown in "timings" row are for Amibroker.com internal use. In development/testing they help us tweak the performance of various parts of analysis engine. The detailed timings do not sum up to actual run time. No further information is available at the moment. The "timings" row is subject to change/removal in the future versions.

If you have problems/questions with the performance of new Analysis window please copy (Ctrl+C) the contents of Info tab and send to support.

- New Analysis: first-phase backtest signal processing and ranking moved to non-UI threads, enabling better parallelism, results in better backtest performance (upto 2x in with trivial codes)

CHANGES FOR VERSION 5.44.0 (as compared to 5.43.1)

- New Analysis: fixed AddToComposite()
- New Analysis: Interval() function was working incorrectly in new analysis window. Fixed.
- New Analysis: Backtest ~~~EQUITY symbol did not have "use only local database" set in 5.43.1. Fixed.
- New Analysis: Backtest performance improved by removing some unnecessary list view refreshes (now it should never be slower than old AA even with simplest formulas)

CHANGES FOR VERSION 5.43.1 (as compared to 5.43.0)

- When range different than "All quotes" was selected, backtest in new Analysis window could produce incorrect results in 5.43.0. Fixed.

CHANGES FOR VERSION 5.43.0 (as compared to 5.42.0)

- **New Analysis Window: multi-threaded Backtest feature implemented**

Note that only first phase of backtest is multithreaded (each symbol in separate thread). 2nd phase (custom backtest) is executed in main thread as there is only one symbol to work on (i.e. equity) and it needs to talk with OLE which is single threaded and GUI.

- Load/Unload buttons removed from Plugins dialog. This was developer-only feature and it was abused by ordinary people who were unloading plugins that were actually in - use (such as data plugins when being connected to given data source)
- ListViews: the last line (partially shown) in RT quote window was not easily selectable. Fixed
- New Analysis window: Implemented Parameters dialog

- AFL: GetCursorMouseButtons() sometimes missed clicks when multithreading was on and formula took ages to execute. Now it is now 100% reliable.
- Auto-selection of proper layer based on selected viewing interval
NOTE: This feature can be turned off by click with RIGHT mouse button over layers window and UNCHECKING "Auto-select layer"
- IRA accounts support - implemented Settlement delay in backtester, via SetOption("SettlementDelay", x)

"SettlementDelay" option describes the number of days (not bars) it takes for sale proceeds to settle and be available for opening new positions.

SetOption("SettlementDelay", 3); // this will cause that proceeds from sale are only available for trading on 3rd day after sale

For detailed tracking

" Detailed log" report option now shows available and unsettled funds for T+1, T+2 and so on

Note: when using this option it is recommended to use backtestRegularRaw instead of backtestRegular, otherwise some trades may not be entered because funds are not settled immediately and you need to be able to enter not on first but subsequent buy signals and that is exactly what backtestRegularRaw offers.

Note2: old backtester (Equity() function) ignores settlement delay

- Zoom in/out toolbar buttons zoom in finer steps (+/-10%)
- Analysis ListView, RMB menu new option: "Add Rank column" - adds a column with ordinal rankings based on current sort or just row number column when list is not sorted
- New feature: Adding new chart/research/account/analysis via (+) tab
- New Analysis window: Implemented SetSortColumns
- New Analysis window: Implemented AddSummaryRows

CHANGES FOR VERSION 5.42.0 (as compared to 5.41.0)

- Implemented right click menu in new Analysis windows (Add symbols to watchlist/Show arrows functionality)
- Zoom in/out toolbar buttons now have auto-repeat feature
- In 5.41.0 Ctrl+C from list view copied text without tabs (column separators). Fixed
- In 5.41.0 did not refresh progress dialog optimization status list. Fixed. (owner draw list view initial size is retrieved in PreSubclass now)
- In 5.41.0 did not display Time&Sales in reverse order (newest on top). Fixed.

CHANGES FOR VERSION 5.41.0 (as compared to 5.40.3)

- **New Analysis window (File->New->Analysis) - featuring multithreaded Scan and Exploration**
Performance note:
to fully benefit from multithreading it is best to use AmiBroker local NATIVE database (not external)
as it is the only database that can deliver data quickly enough.

Tests show that on AmiBroker native databases scans and explorations are 100% scalable to multiple cores
- i.e. for example would run upto 8 times faster on 8 CPU/core machine.

IMPORTANT: This window is work-in-progress. Multithreaded Backtest and Optimization features will be added later.

- AFL: AddColumn - added mini bar charts to explorations
Exploration now features ability to create mini bar charts in individual cells.

AddColumn has new parameter 'barchart'

AddColumn(ARRAY, "Caption", format = 1.2, fgcolor = colorDefault, bkcolor = colorDefault, width = -1, barchart = null)

'barchart' parameter accepts values from 0...100 representing percentage width of bar chart displayed in a cell
the in-cell bar chart is drawn with bkcolor (background color).

Example usage:

```
Filter=1;  
AddColumn( Close, "Close" );  
rank = PercentRank( Close, 100 );  
Color = ColorHSB( rank * 64/100, 255, 255 );  
AddColumn( rank, "100-day percent rank", 1.2, colorDefault, Color, -1, rank );
```

Note that although this example uses same value for numeric display and chart bar width, it does NOT need to be the same, i.e. numerical value of the cell is independent from bar chart.

- Totally rewritten listview with very efficient owner draw code speeds up list view scrolling/resizing/repainting more than 10 times. Native system ListView control is terribly slow on Windows XP/Vista/7 (surprisingly it was many times faster in Win9x!)
- AFL: Percentile() does a param check for range > 0 now
- Mini-chart in RT Quote window - Mini chart shows where LAST price is within LOW-HIGH range.
- Errors and warnings are now displayed in red color in the indicator pane
- During actual GDI chart rendering AmiBroker measures time and if it exceeds 500ms timeout it stops rendering with Warning 901.
This check is required because GDI rendering occurs in GUI thread and if GUI thread does not process messages for half second Windows will think that application is "not responding".
This prevents making application stuck with some overly complex, possibly incorrect formulas that call hundreds of Plot()s
- Warning 502: Calling Plot()/PlotOHLC over 500 times is displayed in indicator in runtime to prevent abuse
- All list views, pressing Ctrl and NumPad '+' automatically adjusts column widths to content

- if data are missing in the in-memory cache, Foreign() calls from non-GUI threads use synchronous retrieval method for reliability.
This provides reliable results in AA but may slow down the multi-threaded operations if cache size is too small and user uses more Foreign() calls than in-memory cache size.

IMPORTANT:

For optimum performance in-memory cache setting (Tools->Preferences->Data) should be GREATER than number of Foreign() calls in any single formula used.

Values lower than that cause lots of disk activity and serialization of all Foreign calls that means that additional threads must wait for data retrieval to complete.

- Old and new Analysis window: outputting millions of rows to listview is much faster now
- All list views - Ctrl+A (Select All) made much faster

HOW TO REPORT BUGS

If you experience any problem with this beta version please send detailed description of the problem (especially the steps needed to reproduce it) to support at [amibroker.com](mailto:support@amibroker.com)