Clarion Handy Tools

CHT TEMPLATES
NEW/REVISED: 2017
(BUILD: 22A.00.00)
GENERATED:
DECEMBER 21, 2017

CATEGORY:
APP Code Signing

*** CODESIGNEXECUTABLE ***

VERSION:	22A.00.00
TYPE:	UTILITY
CATEGORY:	APP Code Signing
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This template when executed runs the MS SignTool wizard against the latest instance of the application .EXE or .DLL, facilitating and automating code signing your executables from the IDE. As of September 2012 an optional addition to this template may be used to copy the finish-signed executable to a target directory on the same drive or on another available drive.

*** HANDYCODESIGNINGEXTENSION ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Code Signing
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED JUNE 2017 ***

This global extension template runs the MS SignTool Tool against your EXEs or DLLs on completion of compile, facilitating and automating code signing your executables at compile time. An option is provided to pre-compress the executable before it is code signed.

HandyCodeSigningExtension is a child of global template AACHTControlPanel and as such, must be installed to the global extensions clarea by first installing AACHTControlPanel and inserting with that template selected. As of September 2012 a new dialog on this template provides for optional capability to have the signed executable copied to an alternate target location (drive and/or directory) after code signing has completed. NOTE TO DEVELOPERS: ABOUT ORDER OF OPERATIONS WHEN CODE SIGNING Since code-signing your executable must take place AFTER other operations such as executable file compression this template now detects when the CHT HandYExecutableCompressor Template is present and enabled, and ensures that the signing operation is performed AFTER compression.

CATEGORY:
APP Configuration

*** AACHTCONTROLPANEL ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Configuration
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	None

DESCRIPTION:

** REVISED DECEMBER 2017 **

This template is a © CHT Global Control Panel. Populating any single CHT template causes this "supervisory" template to appear automatically in the global extensions area of your application. From this interface you can adjust global CHT settings used by other templates in our suite.

AACHTControlPanel also provides useful "state-of-your-application" reporting features. The following global templates are designated as children of AACHTControlPanel, and as such, you must select AACHTControlPanel, before child templates become visible for inclusion in your application. Child templates are: AACHTControlPanelClose, INIFile, VblFileNames, ApplicationImagesEx, HandyVersionResource, HandyNTServiceFunctions HNDCMPControlPanel, ExportClassesByName, ExportClassesByCategory, and FieldLevelValidationCode.

AACHTControlPanel generates and controls a number of equates and compiler pragmas that are important for the proper functioning and linking of our classes into your application. For this reason, all CHT templates check for the existence of this template in your application and re-populate it if it is removed.

*** EMBEDOBJECT ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Configuration
SCOPE:	ABC PROCEDURE
CLASSES:	Any ABC Compliant Class
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED JUNE 2017 ***

This template creates a single, local instance of any class that you select from the standard list of classes available in the "Application Builder Class Viewer". By default the viewer loads and displays any /LIBSRC/-located class that contains the tag !ABCIncludeFile in it's header (.INC) file.

All methods may be embedded (overridden) in the usual fashion. On the "Classes" tab, replace the words "Object Name Here" with an instance name for your object. Then select the class you wish to use in the pulldown.

NOTE: Any special includes such as drivers or LIBs that the selected object may require will have to be added into the project manually using the "Project" button.

*** HANDYVERSIONRESOURCE ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Configuration

SCOPE: APPLICATION
CLASSES: None

PARENT: AACHTControlPanel

DESCRIPTION:

*** REVISED DECEMBER 2017 ***
This global application extension template incorporates standard version information into your executable. It generates an "application.version" file which is linked into the EXE, DLL or LIB. Company name, Copyright, Trademark, Version Number, Build Number, File Information, Language, Locale and custom information are included. This template also interacts with AACHTControlPanel's Version dialog - which produces application constant HPROP:AppVersionEX - to synchronize the version element of the HPROP:AppVersionEX as version values change on this template.

HandyVersionResource is a child of global template AACHTControlPanel and as such, must be installed to the global extensions area by first installing AACHTControlPanel and inserting with that template selected.

*** INIFILE ***

VERSION: 22A.00.00
TYPE: EXTENSION
CATEGORY: APP Configuration
SCOPE: APPLICATION
CLASSES: HNDIni HNDError
PARENT: AACHTControlPanel

DESCRIPTION:

*** REVISED JUNE 2017 ***

This extension extends the capabilities of the ABC IniClass.

** Developers can optionally establish an alternative INI file for app configuration settings written by CHT templates and classes.

** Developers can optionally establish an application INI file that's tied to the name of the application exe, such that when the exe is renamed, even after compilation, the INI takes on the new name of your exectuable file.

** Developers can optionally override the directory location of the application INI file as configured on the ABC global properties dialog.

** Developers may optionally redirect all application configuration I/O that would normally go to an INI file or to the Windows Registry into a dictionary table provided by the application itself. This functionality works in conjunction with another CHT template called INITableIOProcedure which must be created first, and selected on this template interface. It also populates a child template (with no user interface) to all of the procedures in your application. The child template writes code into your Frame procedure in order to reference the frame window and frame thread to two global HNDIni Class properties.

As a result you are able to set messages to the frame window status bar with a call to HIni.SetFrameWindowStatus(). Events can likewise be posted from any procedure in your application to the frame window accept loop with HIni.PostFrameWindowEvent(). This template also provides optional "Run Once" capability. If your application is already running on a given computer, subsequent attempts to run it can be short circuited. See the "Run Once" button elsewhere on this template's interface.

INIFile is a child of global template AACHTControlPanel and as such, must be installed to the global extensions area by first installing AACHTControlPanel and inserting with that template selected.

*** UTILREADSETTINGSFROMCONFIG ***

VERSION: 22A.00.00
TYPE: UTILITY
CATEGORY: APP Configuration
SCOPE: APPLICATION
CLASSES: None

PARENT: None

DESCRIPTION:

*** REVISED MAY 2017 ***

This template is used by CHT templates, specifically EmbedWindowClasses, ApplicationImagesEx and ProcedureImagesEx to determine color and image related settings placed in a CHT configuration file, for use in the application where the "Freeze" setting has not been applied.

CATEGORY:
APP Cosmetics

*** GLOBALTRANSPARENTCONTROLS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Cosmetics
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED MAY 2017 ***

This global extension allows you to control transparency on window and report controls throughout your entire application without having to go back and manually check the "transparent" property on a procedure-by-procedure basis. Individual control types may be controlled separately from this global template. A child template called ProcedureTransparentControls is populated on all qualifying procedure types and provides local override capability.

*** PROCEDURETRANSPARENTCONTROLS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Cosmetics
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED MAY 2017 ***

This procedure extension allows you to control transparency on window and report controls created by your procedure without having to go back and manually check the "transparent" property on a structure-by-structure basis. Individual control types may be controlled separately. This template is a child of global template GlobalTransparentControls which must be populated before this one becomes visible. Global settings may be overridden locally on this template's interface.

CATEGORY:
APP Discovery

*** HNDCMPAPPTEMPLATESTOHTMLBOTH ***

VERSION:	22A.00.00
TYPE:	UTILITY

CATEGORY:	APP Discovery
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This template is used by CHT Compile Manager to generate HTML app discovery information. This template is called from CHT Compile Manager as an entry point to two other utility templates. These are: HNDCmpAppTemplatesToHTMLFile which generates application template discovery information and HNDCmpAppClassesToHtmlFile which generates application classes discovery information.

CATEGORY:
APP Help

*** APPSTARTSPLASHUNTHREADED ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Help
SCOPE:	FRAME
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED OCTOBER 2017 ***

This extension template is dropped on the application frame to start your splash screen earlier, with an option to start threaded or unthreaded. This is intended to replace the splash screen dropdown that appears on the ABC frame template which starts too late and causes the splash to lag and often not display correctly until the application has completed loading. This spash screen implementation loads and displays the splash immediately on application load even before the frame procedure loads. If started unthreaded the application frame only displays when the splash completes, or is clicked by the user.

*** EMBEDSTANDARDWEBPAGEHEADER ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Help
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED OCTOBER 2017 *** This template may be used to build apps that generate HTML Help files as do HNDTPXHT.APP and HNDCLXHT.APP and others. While it's normal to develop style sheets in downloadable files so the the same set of style sheets may be shared via link statements amongst numerous web pages and even different web sites, there are situations where it's more expedient to have the style sheet internal to the HTML document in order to make the help document viewable off-line. This template can be added to HTML generating procedures to either assist in generating both screen and print style sheets internal to HTML help documents such as those created by the example applications cited above. At this point in time, most items of a design and cosmetic nature can be changed on the template interface so that the things you're most likely to want changed are under your direct control. Template changes can be applied in the application's generated output at application generation-time by entering constant values directly on the template or, in the event you want to apply changes at application run-time, using variables that the application looks up elsewhere in a configuation file. Not all settings are fully variable, but all key ones are. If you find this template currently too restricting, please let us know and we'll make all settings fully configurable. This template now generates HTML5 compatible output.

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Help
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED SEPTEMBER 2017 ***

This template may be used to build apps that generate HTML Help files. See HNDBULKEMAILBATCHER.APP. While it's normal to develop style sheets in downloadable files so the the same set of style sheets may be shared via link statements amongst numerous web pages and even different web sites, there are situations where it's more expedient to have the style sheet internal to the HTML document in order to make the help document or HTML email viewable off-line. This template can be added to HTML generating procedures to either assist in generating a screen style sheet that is internal to HTML help documents and HTML emails.

At this point in time, most items of a design and cosmetic nature can be changed on the template interface so that the things you're most likely to want changed are under your direct control. Template changes can be applied in the application's generated output at application generation-time by entering constant values directly on the template or, in the event you want to apply changes at application run-time, using variables that the application looks up elsewhere in a configuation file. Not all settings are fully variable, but all key ones are. If you find this template currently too restricting, please let us know and we'll make all settings fully configurable. This template now generates HTML5 compatible output.

Some example applications are: HNDXML2HTMLGEN.APP, HNDDOCUMENTBUILDER.APP, HNDBULKNETMAILSMTP.APP

CATEGORY:
APP Styling

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	APP Styling
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

© CHT ApplicationImagesEx establishes global level application-wide styling settings including wallpapers, icons, colors, fonts and common control properties. All globally applied application settings established here, may be overridden on the "extensions" menu of your window procedures on a child template called ProcedureImagesEx. ApplicationImagesEx provides several "stylesheet" interfaces that, when completed, may be saved to a "master configuration" which can then be read into other applications.

Aside from reducing the drudgery of having to style (wallpaper, icons, colors, fonts, common control properties) on every window in every app, ApplicationImagesEx helps to make your application suites stylistically consistent. While applications can be "style-frozen" to make them unique from the master configuration, developers can, by simply "unfreezing" an application's stylistic settings it can immediately be restyled to the currently configured standard.

ApplicationImagesEx can track the images used in your application, list them in a special IMAGES file, and even copy images scattered around your drive to the standard images directory or to a directory provided on the template. This makes the task of moving an application to another machine really easy since the images it uses are all in one place.

ApplicationImagesEx can be made to control display settings such as fonts, colors, charset, button styles, prompt styles, entry field background color, and more.

ApplicationImagesEx includes and makes available the icons required when the EmbedWindowFunctions template implements CHT extended menus. This template, can if necessary, be disabled so as to eliminate any stylistic impact it has on your application while still supporting other CHT templates that from-time-to time require it's services for other than stylistic reasons.

ApplicationImagesEx is a child of global template AACHTControlPanel and as such, must be installed to the global extensions area by first installing AACHTControlPanel and inserting with that template selected.

ApplicationImagesEx also includes an optional global switch which when set, causes the template to generate code for Clarion-10-style extended menus in all application windows, not just the FRAME window. The ABC templates provide only for extended (colored) menus in the master FRAME window.

CATEGORY:
APP Versioning

*** NEW APP MAIN FRAME PROCEDURE #1 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** REVISED DECEMBER 2017 *** This procedure provides a starting MAIN MDI FRAME that can be quickly modified with your own specifics to make a container frame for your new application and implements CHT templates AACHTControlPanel, ApplicationImagesEx, HandyVersionResource, and EmbedWindowFunctions to provide standard services such as dynamic icon menus, run-once, application-in-tray, application versioning and more.

*** NEW APP MAIN FRAME PROCEDURE #2 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides a starting MAIN MDI FRAME that can be quickly modified with your own specifics to make a container frame for your new application and implements CHT templates AACHTControlPanel, ApplicationImagesEx, HandyVersionResource, and EmbedWindowFunctions to provide standard services such as dynamic icon menus, run-once, application-in-tray, application versioning and more.

*** NEW APP MAIN FRAME PROCEDURE #3 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various

PARENT: Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides a starting MAIN MDI FRAME that can be quickly modified with your own specifics to make a container frame for your new application and implements CHT templates AACHTControlPanel, ApplicationImagesEx, HandyVersionResource, and EmbedWindowFunctions to provide standard services such as dynamic icon menus, run-once, application-in-tray, application versioning and more.

*** NEW APP MAIN FRAME PROCEDURE #4 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides a starting MAIN MDI FRAME that can be quickly modified with your own specifics to make a container frame for your new application and implements CHT templates AACHTControlPanel, ApplicationImagesEx, HandyVersionResource, and EmbedWindowFunctions to provide standard services such as dynamic icon menus, run-once, application-in-tray, application versioning and more.

*** NEW APP MAIN FRAME PROCEDURE #5 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides a starting MAIN MDI FRAME that can be quickly modified with your own specifics to make a container frame for your new application and implements CHT templates AACHTControlPanel, ApplicationImagesEx, HandyVersionResource, and EmbedWindowFunctions to provide standard services such as dynamic icon menus, run-once, application-in-tray, application versioning and more.

*** NEW APP MAIN FRAME PROCEDURE #6 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides a starting MAIN MDI FRAME that can be quickly modified with your own specifics to make a container frame for your new application and implements CHT templates AACHTControlPanel, ApplicationImagesEx, HandyVersionResource, and EmbedWindowFunctions to provide standard services such as dynamic icon menus, run-once, application-in-tray, application versioning and more.

*** NEW SPLASH SCREEN #1 ***

VERSION: 22A.00.00
TYPE: JUMPSTART
CATEGORY: APP Versioning
SCOPE: PROCEDURE
CLASSES: Various
PARENT: Various

DESCRIPTION:

*** REVISED DECEMBER 2017 *** This procedure provides an example splash screen that can be quickly modified with your own specifics to make a splash for your own applications. This example implements several CHT templates that let your splash provide up-to-date build information about your application as you recompile and re-release.

*** NEW SPLASH SCREEN #2 ***

VERSION: 22A.00.00
TYPE: JUMPSTART
CATEGORY: APP Versioning
SCOPE: PROCEDURE
CLASSES: Various
PARENT: Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides an example splash screen that can be quickly modified with your own specifics to make a splash for your own applications. This example implements several CHT templates that let your splash provide up-to-date build information about your application as you recompile and re-release.

*** NEW SPLASH SCREEN #3 ***

VERSION: 22A.00.00
TYPE: JUMPSTART
CATEGORY: APP Versioning
SCOPE: PROCEDURE
CLASSES: Various
PARENT: Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides an example splash screen that can be quickly modified with your own specifics to make a splash for your own applications. This example implements several CHT templates that let your splash provide up-to-date build information about your application as you recompile and re-release.

*** NEW SPLASH SCREEN #4 ***

VERSION: 22A.00.00
TYPE: JUMPSTART
CATEGORY: APP Versioning
SCOPE: PROCEDURE
CLASSES: Various
PARENT: Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides an example splash screen that can be quickly modified with your own specifics to make a splash for your own applications. This example implements several CHT templates that let your splash provide up-to-date build information about your application as you recompile and re-release.

*** NEW SPLASH SCREEN #5 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides an example splash screen that can be quickly modified with your own specifics to make a splash for your own applications. This example implements several CHT templates that let your splash provide up-to-date build information about your application as you recompile and re-release.

*** NEW SPLASH SCREEN #6 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** NEW DECEMBER 2017 *** This procedure provides an example splash screen that can be quickly modified with your own specifics to make a splash for your own applications. This example implements several CHT templates that let your splash provide up-to-date build information about your application as you recompile and re-release.

*** NEW SPLASH SCREEN #7 ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** REVISED DECEMBER 2017 *** This procedure provides an example splash screen that can be quickly modified with your own specifics to make a splash for your own applications. This example implements several CHT templates that let your splash provide up-to-date build information about your application as you recompile and re-release. A Microsoft I.E Browser control is populated.

*** NEW SPLASH SCREEN #8 ***

VERSION:	22A.00.00

TYPE:	JUMPSTART
CATEGORY:	APP Versioning
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** REVISED DECEMBER 2017 *** This procedure provides an example splash screen that can be quickly modified with your own specifics to make a splash for your own applications. This example implements several CHT templates that let your splash provide up-to-date build information about your application as you recompile and re-release.

CATEGORY:
Browse Extensions

*** ACTIVELINKBROWSECOLUMN_EWF ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Browse Extensions
SCOPE:	WINDOW
CLASSES:	HNDWindow
PARENT:	EmbedWindowFunctions

DESCRIPTION:

*** REVISED JANUARY 2017 ***

This control enables an active startup link in one of your browse columns. The browse may be any ABC browse or any CHT-template-extended browse. It requires that the EmbedWindowFunctions template be placed on the window first.

When a valid Web-Url is found in the row being clicked, clicking the browse column enabled by this template starts your default Internet browser enabling you to jump to the web site found there.

Similarly when a valid Email-Address is found in the row being clicked, clicking the browse column enabled by this template starts your default email client, enabling you to create an email message to the address found there.

Finally, when the correctly pathed name of an executable is found in the row being clicked, clicking the browse column enabled by this template starts (runs) that application similarly to starting the application from a menu.

This template shares an embeddable instance of the HNDWindow Classes. This template is a child template of CHT EmbedWindowFunctions and as such, that template must be present before this one can be added.

*** BROWSELESSEDITFORM ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Browse Extensions
SCOPE:	WINDOW
CLASSES:	None
PARENT:	BrowseBox(ABC)

DESCRIPTION:

*** REVISED OCT 2017 ***

This template lets you open an edit form on your data table without having to use a browse ahead of it. The template populates three buttons: Insert, Previous and Next. Insert lets you add new records while Previous and Next let you navigate the table records. The data table requires no keys although tables with keys may be used. The form normally opens looking at the physically-first record in the table. You are able to set an optional switch on the template interface so that the form can, in fact, be used with a browse. In this case the first record displayed is the record currently selected in your browse. Once in the form, records may be navigated forward and back and multiple records may be inserted in succession. At EOF and BOF you are given a message indicating that beginning or end of file have been reached.

*** EMBEDVIEW_LBX ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Browse Extensions
SCOPE:	WINDOW PROCEDURE
CLASSES:	None
PARENT:	ListBoxBrowseExtender

DESCRIPTION:

*** REVISED JUNE 2017 ***

The _LBX prefix in this template name indicates that it is a child template associated only with (L)istbox(B)rowseE(X)tender. This control template generates a Multi-Table view with an accompanying ABC ViewManager for tables added to the Tables Schematic *(provided by this template)* area of your ListBoxBrowseExtender procedure. EmbedView_LBX appears only in your list of available control templates when control template, ListBoxBrowseExtender is first added to your window. It generates VIEW processing code into the `HNDFileLoadBrowse.FillFromView` placeholder procedure which is able to fill the ListBoxBrowseExtender queue with records from multiple tables as laid out in the file table tree that this template addes to your procedure. The Clarion VIEW structure created along with the ABC View Manager instantiated allows your app to interact with multiple tables as if they were one, big flat file, as joins between tables are handled internally by the VIEW structure.

*** HANDYFILEQUEUEBUILDER ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Browse Extensions
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	Window(ABC), Report(ABC), Process(ABC)

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This template builds a queue based on file (table) definitions in your dictionary. Drop it on any procedure that has tables defined in the procedure tables schematic. To generate a queue based on table fields, complete the "File/Queue Information tab by selecting the fields that should be populated to your queue. A queue created by this template can be used with CHT template ListBoxBrowseExtender to build a fully functional queue based browse independent of any ABC browse templates. Note that the template now warns you that local variables are not permitted to be placed into the queue. As of December 2008, this template now also works with ABC Source procedures.

*** MAXPRELOADRECORDCONTROL_LBX ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Browse Extensions
SCOPE:	WINDOW PROCEDURE
CLASSES:	None
PARENT:	UniversalQueryControl_LBX

DESCRIPTION:

*** REVISED JUNE 2017 ***

The _LBX suffix on this template name indicates that it is a child template associated only with (L)istbox(B)rowseE(X)tender. This control template drops two controls, a prompt and a spin-number control onto your LBX browse window. The spin-number control is used to increase or decrease the "LbxMaxLoadTimeRecords" variable created by ListBoxBrowseExtender. filling it with all available records. When LBX is set to "FillFromCHTServer" this "LbxMaxLoadTimeRecords" controls the number of records initially pre-loaded into the MAIN queue. Other than positioning and/or cosmetically altering these controls from the IDE window editor [optional] no further configurations are necessary on this template interface. NOTE:This template is a child of UniversalQueryControl_LBX so it will not appear in the Clarion ToolBox of selectable Control Templates unless, UniversalQueryControl_LBX is already present on the window.

*** PREFILTERBUTTONS_LBX ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Browse Extensions
SCOPE:	WINDOW PROCEDURE
CLASSES:	None
PARENT:	ListBoxBrowseExtender

DESCRIPTION:

*** REVISED JUNE 2017 ***

The _LBX suffix on this template name indicates that it is a child template associated only with (L)istbox(B)rowseE(X)tender. This control template drops two buttons (Clear & Refresh), (Set & Refresh) onto your LBX browse window. The "Clear & Refresh" button clears the load-time browse pre-filter and refreshes the browse and filling it with all available records. The "Set & Refresh" re-establishes the load-time browse pre-filter and refreshes the browse, thus filling with only those records that match this pre-filter. Other than positioning and/or cosmetically altering the buttons from the IDE window editor [optional] no further configurations are necessary on this template interface.

CATEGORY:
CHT Snap-In

*** APPLICATIONSNAPINS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	CHT Snap-In
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED DEC 2017 ***

This global extension template is an incorporation point in your application for a number of stand-alone, mostly-single-function executables which have, by and large, been a growing and ever-evolving part of CHT's `HNDAPPS` suite of demo and utility applications for many years.

At this time, CHT has provided Snap-Ins for HTTP file movement, File compression and decompression, EMAIL protocols of various sorts, File Encryption and End User Support.

The individual "Snap-In" items listed and described below are the components which are fully available to you with any given CHT Build or Build Update. This list continues to evolve over time as this template's interface continues to evolve.

• Snap-Ins Presently Available Via This Template: (6)

CHTSNAPSEND.EXE, CHTSNAPEDIT.EXE, CHTSNAPGET.EXE, CHTSNAPMAPI.EXE,
CHTSNAPSMTP.EXE, CHTSNAPZIP.EXE AND CHTSNAP2PDF.EXE

CHT "Snap-Ins" are written in C#. The source code for these is available separately (priced individually) by
special request.

• CHT Batch-Bots Presently Available Via This Template: (10)

HNDMAKEHZO.EXE, HNDMAILMAPI.EXE, HNDMAILCDO.EXE, HNDMAILSMTP.EXE
HNDSEARCH.EXE, HNDSLFCMD.EXE, HNDMAILNET.EXE, HNDZIPNET.EXE,
HNDENCRYPT.EXE,
HNDFTPNET.EXE and HNDXML2HTMGEN.EXE

CHT "Batch-Bots" are written in Clarion. The source code for these is available in your `HNDAPPS` directory
as a standard part of your CHT toolkit.

CHT "Batch-Bots" and CHT "Snap-Ins" are primarily command-line utilities meant to perform a specific
task independent of a standard windows-form interface.
Some examples are: zipping a file (CHTSNAPZIP.EXE), or sending a file to a web-server (HNDSLFCMD.EXE)
or fetching a file from a web-server (CHTSNAPGET.EXE), and so on.

These executables, while fully stand-alone, so they can be used from the DOS command-line, are also
viable application components used as you would a DLL, so they can be run also from inside a Clarion or
C# application. Each runs as an independent process spawned from inside your application. Specific
instructions as to how to perform any individual task are passed to these executables, on the command-
line.

For example: `CHTSNAPZIP zz MYAPP.ZIP c:\mydir\myapp.exe`
This set of commandline instructions, creates a zip called "MYAPP.ZIP" from an EXE called MYAPP.EXE.

• CHT "Utility" Applications Now Available Via This Template: (4)

CHT's `Application Snap-Ins` template presently provides in-app launch capability for these "Utility
Applications":
HNDPRINTHTML.EXE, HNDPREVIEWER.EXE, HNDSETUP.EXE, HNDSQLTRACE.EXE

CHT "Utility Executables" are written in Clarion. The source code for these is also available, like "Batch-
Bots", in your `HNDAPPS` directory as a standard part of the CHT Toolkit. CHT "Utility Executables" differ
from "Snap-Ins" and "Batch-Bots" in that they have full-blown Windows-Form interfaces from which
configuration and task instructions are requested from the application. They generally don't need to be
launched *though some can be* with command-line instructions.

*** EMBEDSNAPCALL ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	CHT Snap-In
SCOPE:	ABC PROCEDURE
CLASSES:	HNDSnapCall
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED JUNE 2017 ***

This template may be used to attach to your application, at the procedure level, a CHT Code module
containing a single class called HNDSnapCall. This extension creates a single, local instance of the
HNDSnapCall Class which is used to call a variety of "CHT SNAP-IN" executable tools in a manner that
resembles standard DLL calls, using parameter passing on the command line.

At time of writing, MARCH 2015, this class interfaces to the following items: CHTSNAPSEND.EXE,
CHTSNAPEDIT.EXE, CHTSNAPSMTP.EXE, CHTSNAPMAPI.EXE, CHTSNAPGET.EXE, CHTSNAPGETEX.EXE
Future: CHTSNAPFTP.EXE, CHTSNAPSMS.EXE, CHTSNAPSFTP.EXE, All methods appear in the "Local
Objects" section of the procedure to which this extension is applied and may be embedded in the usual
ABC fashion.

*** GLOBALINCLUDEHNDSNAPCALLFUNCTIONS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	CHT Snap-In
SCOPE:	APPLICATION
CLASSES:	HNDSnapCall
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED JUNE 2017 ***

This extension template globally instantiates a single instance of the CHT HNDSnapCall class. You may change the default instance name if you wish. This template is a child of AACHTControlPanel and can only be populated to your application once AACHTControlPanel has been added.

NOTE: This template is auto-populated (once) to the global area of your application, when any one of our "CHT Snap" functionalities is enabled on the global template called Application Snap-Ins. Since many snap-call functions include RUN() calls with the console output redirected for capture, into an application variable, this template suggests optionally that the application copy hndcapcon.dll to the application compile output directory.

*** RUNCHTSNAPEDIT ***

VERSION:	22A.00.00
TYPE:	CODE
CATEGORY:	CHT Snap-In
SCOPE:	ABC PROCEDURE
CLASSES:	HNDSNAPCALL.CLW
PARENT:	ApplicationSnapIns

DESCRIPTION:

*** REVISED JUNE 2017 ***

This code template helps you to write code that starts CHTSNAPEDIT.EXE from a Clarion menu or button control ACCEPT event. This code template is dependent on the initial placement of the global CHT procedure-local template EmbedSnapCall as well as the CHT global templates ApplicationSnapIns and its parent AACHTControlPanel. In fact, this code template will not appear in your list of CHT code templates unless AACHTControlPanel, ApplicationSnapIns and EmbedSnapCall are installed first.

*** SNAPGETSTARTBUTTON ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	CHT Snap-In
SCOPE:	WINDOW
CLASSES:	None
PARENT:	ApplicationSnapIns , Window(ABC)

DESCRIPTION:

*** REVISED JUNE 2017 ***

This control template places a single button on your window that may be used to start the CHT SnapGet Utility. This template is a child of CHT template ApplicationSnapIns(HndTools) which must first be available on the application before this extended control becomes visible for selection and placement on your procedure window. Global template ApplicationSnapins(HndTools), once added to your application, must also have its "Enable?" and "Copy" switches checked, in the CHT SNAPGET dialog. This button control template specifically calls the HNDSNAPCALL.RunCHTSnapGet() function fetch a remote file from either a CHT-Style server or a standard IIS (WIndows) or Linux Server.

Example application HNDTESTCHTSNAPGET.APP.

*** SNAPHNDMAILCDOSTARTBUTTON ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	CHT Snap-In
SCOPE:	WINDOW
CLASSES:	None
PARENT:	ApplicationSnapIns , Window(ABC)

DESCRIPTION:

*** REVISED JUNE 2017 ***

This control template places a single button on your window that may be used to start the CHT HNDMAILCDO.EXE Batch-Bot. This template is a child of CHT template ApplicationSnapIns(HndTools) which must first be available on the application before this extended control becomes visible for selection and placement on your procedure window. Global template ApplicationSnapins(HndTools), once added to your application, must also have its "Enable?" and "Copy" switches checked, in the HNDMAILCDO dialog. This button control template specifically calls the HNDSNAPCALL.RunHndMailCDO() function send an email via HNDMAILCDO.EXE. Example application: HNDTESTMAILCDO.APP.

*** SNAPHNDMAILMAPISTARTBUTTON ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	CHT Snap-In
SCOPE:	WINDOW
CLASSES:	None
PARENT:	ApplicationSnapIns , Window(ABC)

DESCRIPTION:

*** REVISED JUNE 2017 ***

This control template places a single button on your window that may be used to start the CHT HNDMAILMAPI.EXE Batch-Bot. This template is a child of CHT template ApplicationSnapIns(HndTools) which must first be available on the application before this extended control becomes visible for selection and placement on your procedure window. Global template ApplicationSnapins(HndTools), once added to your application, must also have its "Enable?" and "Copy" switches checked, in the HNDMAILMAPI dialog. This button control template specifically calls the HNDSNAPCALL.RunHndMaiLMapi() function send an email via HNDMAILMAPI.EXE. Example applications: HNDZIPNEMAILBBOTMAPI.APP, HNDTESTMAILMAPI.APP.

*** SNAPHNDMAILNETSTARTBUTTON ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	CHT Snap-In
SCOPE:	WINDOW
CLASSES:	None
PARENT:	ApplicationSnapIns , Window(ABC)

DESCRIPTION:

*** REVISED JUNE 2017 ***

This control template places a single button on your window that may be used to start the CHT HNDMAILNET.EXE Batch-Bot. This template is a child of CHT template ApplicationSnapIns(HndTools) which must first be available on the application before this extended control becomes visible for selection and placement on your procedure window. Global template ApplicationSnapins(HndTools), once added to your application, must also have its "Enable?" and "Copy" switches checked, in the HNDMAILNET dialog. This button control template specifically calls the HNDSNAPCALL.RunHndMailNET() function send an

email via HNDMAILNET.EXE. Example application: HNDTESTMAILNET.APP.

*** SNAPHNDMAILSMTPSTARTBUTTON ***

VERSION: 22A.00.00
TYPE: CONTROL
CATEGORY: CHT Snap-In
SCOPE: WINDOW
CLASSES: None
PARENT: ApplicationSnapIns , Window(ABC)

DESCRIPTION:

*** REVISED JUNE 2017 ***

This control template places a single button on your window that may be used to start the CHT HNDMAILSMTP.EXE Batch-Bot. This template is a child of CHT template ApplicationSnapIns(HndTools) which must first be available on the application before this extended control becomes visible for selection and placement on your procedure window. Global template ApplicationSnapins(HndTools), once added to your application, must also have its "Enable?" and "Copy" switches checked, in the HNDMAILSMTP dialog. This button control template specifically calls the HNDSNAPCALL.RunHndMailSMTP() function send an email via HNDMAISMTP.EXE. Example application: HNDMAILSMTPSNAPTEST.APP, HNDZIPNEMAILBBOT.APP, HNDTESTMAILSMTP.APP

CATEGORY:
Capture DOS Console

*** EMBEDHNDCAPTURECONSOLE ***

VERSION: 22A.00.00
TYPE: EXTENSION
CATEGORY: Capture DOS Console
SCOPE: ABC PROCEDURE
CLASSES: HNDCaptureConsole
PARENT: Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED JUNE 2017 ***

This extension creates a single, local instance of the HNDCaptureConsole Classes to your procedure. It enables your app to perform a RUN() via HNDCaptureConsole.Run which will become "captured" in such a way as to direct all output (including errors) back to your application via the HNDCaptureConsole.Run return value.

Since HNDCaptureConsole functionality is based in CHT DLL HNDCAPCON.DLL, this class requires you to ship HNDCAPCON.DLL with your application. Use of this template ensures that the standard ShipList includes a notation to that effect. The template provides a switch that configures your app to copy HNDCAPCON.DLL to your executable output directory as further configured in your .RED settings.

See HNDBATCHRUNNER.APP, HNDCODESIGN.APP, for demonstrations of this template in action. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion.

CATEGORY:
Client/Server

*** CLIENTSERVERBROWSEBUILDER_XCL ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Client/Server
SCOPE:	ABC PROCEDURE
CLASSES:	HNDHttp HNDClient
PARENT:	ClientServerBrowseListBox

DESCRIPTION:

*** REVISED JUNE 2017 ***

This template extension works as a child of ClientServerBrowseListBox to produce a Client Server Data Browse based on data requested from a Remote CHT Client Server Application which contains view procedures built with the CHT Template ClientServerDataBuilder. This special _XCL version of ClientServerBrowseBuilder assumes an external instance of the CHT HNDClient Class is passed into this procedure from another window which has already instantiated and logged-in the client application to the server being accessed. This procedure instantiates HNDClientHook, which receives and references the passed in HNDClient instance and extends it with several methods containing code with local access to the browse queue, the list box and to a special queue called MacroQ which, optionally, may receive query macro definitions created in the server configuration table. Creating a "Client Server Web Browse" is a snap using this template in conjunction with ClientServerBrowseListBox, by simply matching one-to-one the query-friendly field names being passed up from the host server with the queue fields allocated to the list box. It is not necessary to populate the client browse (and its queue) with all browse-directed fields passed up from the server. At the same time the client browse query control (built with QueryParsingFilterControl) is still able to query all fields enabled for querying in the server view to which this browse connects, regardless of the number of fields displayed in the browse.

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Client/Server
SCOPE:	APPLICATION
CLASSES:	HNDClient HNDClientHook
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED MARCH 2017 ***

This extension template globally instantiates a single instance of the CHT HNDClient class. You may change the default instance name if you wish. This template is a child of AACHTControlPanel and can only be populated to your application once AACHTControlPanel has been added.

VERSION:	22A.00.00
TYPE:	PROCEDURE
CATEGORY:	Client/Server
SCOPE:	WINDOW
CLASSES:	HNDHttp HNDClient HNDClientHook
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED JANUARY 2017 ***
This template builds an HTTP client browse that can access a CHT server back-end serving up any Clarion compatible data table whether ISAM or SQL. This template differs from HandyWebClientBrowseProcedure, only in the query control being auto-populated. This "Ex" version of the template brings with it a newer, "snazzier" query control based in PopFavoriteQueries_CSBL, where the "CSBL" stands for *Client Server Browse ListBox*.
A snap-in client form template HandyWebClientFormProcedure, a snap-in client report template HandyWebClientReportProcedure and a snap-in client process procedure HandyWebClientProcessProcedure provide a full complement of Web Client procedure types. With this

template and it's accompanying snap-ins, building a fast, reliable across-the-web Client/Server application accessing a very ordinary computer using a very standard DSL connection is no more difficult than building an regular Clarion desk-top application. This technology is considerably faster than Clarion's IP driver and it natively incorporates data compression, encryption, back-end views and CHT's query language system. An example client application is provided (HNDMTSCL.APP) which accesses CHT's forum data tables using an example server application called HNDMTSSV.APP. Other example clients are: HNDCLIENTCLLEAN1VIEW.APP,HNDCLIENTCLLEAN4VIEW.APP.

*** HANDYWEBCLIENTUPDATEFORMPROCEDURE ***

VERSION:	22A.00.00
TYPE:	PROCEDURE
CATEGORY:	Client/Server
SCOPE:	WINDOW
CLASSES:	HNDClient HNDHttp HNDClientHook
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED FEB 2017 ***

This template builds a Web Client-Server Update Form that can access a CHT server back-end serving up any Clarion compatible data table whether ISAM or SQL. With this template, building a fast, reliable across-the-web update form accessing a very ordinary computer using a very standard DSL connection is no more difficult than building an regular Clarion desk-top application. This technology is considerably faster than Clarion's IP driver and it natively incorporates data compression, encryption, back-end views and CHT's query language system. An example client application is provided (HNDMTSCL.APP) which accesses CHT's forum data tables using an example server application called HNDMTSSV.APP.

*** HANDYWEBCLIENTUPDATEFORMPROCEDURE_LBX ***

VERSION:	22A.00.00
TYPE:	PROCEDURE
CATEGORY:	Client/Server
SCOPE:	WINDOW
CLASSES:	HNDClient HNDHttp
PARENT:	Window(ABC)

DESCRIPTION:

*** NEW MARCH 2017 ***

This template builds a Web Client-Server Update Form that can access a CHT server back-end serving up any Clarion compatible data table whether ISAM or SQL. With this template, building a fast, reliable across-the-web update form accessing a very ordinary computer using a very standard DSL connection is no more difficult than building an regular Clarion desk-top application. This technology is considerably faster than Clarion's IP driver and it natively incorporates data compression, encryption, back-end views and CHT's query language system. An example client application is provided (HNDMTSCL.APP) which accesses CHT's forum data tables using an example server application called HNDMTSSV.APP.

CATEGORY:
Client/Server Browse

*** CLIENTSERVERBROWSELISTBOX ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Client/Server Browse
SCOPE:	ABC PROCEDURE
CLASSES:	HNDFileLoadBrowse

PARENT: Window(ABC)

DESCRIPTION:

*** REVISED AUGUST 2017 ***

This control template populates a list box control, a set of edit buttons and the CHT classes and code necessary to build a Clarion browse that populates from a remote web server, like demo application HNDMTSSV.APP. A child template called ClientServerBrowseBuilder_XCL completes ClientServerBrowseListBox and is also required to be added to your procedure. ClientServerBrowseListBox can import the field names designated on your client server application as web browse fields in order to build a background queue that underlines the list box. This template is also compatible with QueryParsingFilterControl so that you can create CHT Query Language queries on your client window which are completed by your remote web-server application. For an example application see HNDMTSCL.APP which illustrates and provides Client-Server access to CHT's Subscriber Support Forum. The demo server servicing HNDMTSCL.APP is called HNDMTSSV.APP and is also available as a CHT demonstration application in the /hndapps/ directory. A document called CLIENT SERVER EXAMPLES explains how to add a server VIEW into HNDMTSSV.APP and a client browse to consume that VIEW into HNDMTSCL.APP.

An example server with 2 views and all server plumbing in place is called: HNDMTSSV.APP An example client with 2 browses but all client plumbing in place is called: HNDMTSCL.APP

*** CLIENTSERVERUPDATEFORMBUTTONS ***

VERSION: 22A.00.00
TYPE: EXTENSION
CATEGORY: Client/Server Browse
SCOPE: ABC PROCEDURE
CLASSES: HNDClientHook
PARENT: HandyWebClientUpdateFormProcedure

DESCRIPTION:

*** REVISED MARCH 2017 ***

This control template populates a set of update controls to a procedure created with CHT Template HandyWebClientUpdateFormProcedure. This template assists with importation of the a back end view created on your server application using CHT templates HandyJDODataAdapter and ClientServerDataBuilder. It also places the HNDClientHook class and references it to a passed-in instance of HNDHttp and writes all I/O code for fetching a requested record, posting it back and/or deleting it. For an example server application see HNDMTSSV.APP. This template, then completes the functionality of a HandyWebClientUpdateFormProcedure and provides controls that save, save and continue, refresh from server, and cancel the procedure window. For an example application see HNDMTSCL.APP which illustrates and provides Client-Server access to CHT's Subscriber Support Forum. The demo server servicing HNDMTSCL.APP is called HNDMTSSV.APP and is also available as a CHT demonstration application in the /hndapps/ directory. This template is a child of CHT Template HandyWebClientUpdateFormProcedure, and as such that template must be applied before this one is avaialable to be added. As of March 2017, hhis template is also a child of CHT Template HandyWebClientUpdateFormProcedure_LBX which creates a Client Server Web Form Procedure compatible with ListBoxBrowseExtender.

*** LISTBOXBROWSEEXTENDER ***

VERSION: 22A.00.00
TYPE: CONTROL
CATEGORY: Client/Server Browse
SCOPE: WINDOW PROCEDURE
CLASSES: HNDFileLoadBrowse
PARENT: Window(ABC)

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

This template automatically gives your queue-based listboxes extended characteristics such as Windows-style record marking with shift-click for marking ranges, ctrl-click for toggling and single-click for marking individual records. Full keyboard-only marking support is also provided.

SUMMARY OF KEYSTROKES USED FOR IN-QUEUE RECORD MARKING

- MouseLeft - Mark record in single record mode.
- ShiftMouseLeft - Mark end of range between selected and current row.
- CtrlShiftMouseLeft - Continues marking from selected to current row.
- CtrlMouseLeft - Toggle record marked/unmarked.
- CtrlSpace - Same effect as CtrlMouseLeft.
- ShiftUp - Mark next record up from selected.
- ShiftDown - Mark next record down from selected.
- UpKey - Switch to single record mode and move cursor up.
- DownKey - Switch to single record mode and move cusror down.
- CtrlUp - Move cursor up with no marking or unmarking.
- CtrlDown - Move cursor dn with no marking or unmarking.
- ShiftPgDn - Mark all from selected to top of current page.
- ShiftPgUp - Mark all from selected to end of current page.
- ShiftHome - Mark all from selected to first record.
- ShiftEnd - Mark all from selected to last record.
- HomeKey - Unmark all and select first record.
- CtrlHome - Unmark all and select first record.
- EndKey - Unmark all and select last record.
- CtrlEnd - Unmark all and select first record.
- PgUpKey - Unmark all and select last record on page.
- PgDnKey - Unmark all and select first record on page.
- CtrlA - Select all.
- CtrlH - "H" how many selected.

The template adds "Explorer-like" features to your listboxes including column-header-click sort orders, reverse sortation and more. For FAST ABC-Browse-Template-FREE browses with local data or remote data this template is your ticket to browse data location freedom! To create a from-scratch ListBoxBrowsExtender procedure that includes a window and a listbox ready for configuration, use the procedure template called "ListBox BrowseExtender Client Server Browse".

Some example applications are: HNDPEOPLE_LBX.APP, HNDLBXDM.APP and HNDSCHOOL.APP.

*** LISTBOXBROWSEEXTENDERCLIENTSERVERBROWSE ***

VERSION:	22A.00.00
TYPE:	PROCEDURE
CATEGORY:	Client/Server Browse
SCOPE:	WINDOW
CLASSES:	HNDHttp HNDClient
PARENT:	ListBoxBrowseExtender

DESCRIPTION:

*** NEW JANUARY 2017 ***

This template builds a Client/Server ListBoxBrowseExtender (LBX) Client/Server browse that works with local data like an ABC browse *AND* can also access a CHT server back-end serving up any Clarion compatible data table whether ISAM or SQL. This template assumes that globally your application has applied at least these global templates: AACHTControlPanel and VBLFileNames.

Once a procedure is created with this template follow these steps:

- STEP 1: Make sure that AACHTControlPanel and VBLFileNames are attached globally.
- STEP 2: Attach tables to your new procedure from the Clarion IDE's DATA/TABLES tab.
- STEP 3: Open the ResizeHandlesControl dialog and insert the "?Display" control into the dropdown.
- STEP 4: Open the LBX template dialog "ListBox Queue" and select a primary browse table.
- STEP 5: On the same dialog, add queue fields to the "Queue Fields" dialog in the order you want

them displayed on the browse.
- STEP 6: Visit the WINDOW PROPERTIES button on your procedure and find the "?List1" Control

Example applications are: HNDPEOPLE_LBX.APP, HNDLBXDM.APP, HNDLBXEXAMPLE.APP

CATEGORY:
Coding Assistance

*** CHTBESTPRACTICEEMBEDCLASS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Coding Assistance
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED JUNE 2017 ***

This template assists the developer with "Best Practice" embedding as discussed in several CHT papers written by Gus M. Creces concerning the topic of Clarion embedding practices. These papers may be found here:

Clarion Embedding Best Practice - Paper One

Clarion Embedding Best Practice - Paper Two

Clarion Embedding Best Practice - Paper Three

Example applications:

HNDBRWCL.APP, HNDBACKUPCONFIG.APP HNDCODESIGN.APP, HNDSMTPNETSEMO.APP, HNDNETZIPDEMO.APP HNDSQLTRACE.APP, HNDBATCHRUNNER.APP, HNDBULKMAILBATCHER.APP

CATEGORY:
Compile Manager

*** HNDCMPCONTROLPANEL ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Compile Manager
SCOPE:	APPLICATION
CLASSES:	HNDClarionDDE
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This global extension template enables your application to be controlled from the CHT Compile Manager. The settings listed below may be changed on an application-level basis from the compile manager control panel.

1) Control EXE, DLL application extension on=DLL off = EXE

2) Control DLL load address

3) Control Library or DLL Link mode on=Library, off=DLL

4) Control debug mode compile on/off

5) Control expire mode compile on/off

6) Control Translation file (English default, or Developer created) on/off

7) Control Processor Affinity on/off

8) Control Code signing on/off

9) Control TXA export on/off during compile pass

10) Control executable compression on/off

11) Control copying of completed exe/dll to configured target

12) Contol Stop On Error on/off

13) Contol HTML reporting on/off

14) Display success or failure of individual app compiles

HNDCMPControlPanel is a child of global template AACHTControlPanel and as such, must be installed to the global extensions area by first installing AACHTControlPanel and inserting with that template selected.

*** HNDCMPIMPORTAACTLPANEL ***

VERSION: 22A.00.00
TYPE: UTILITY
CATEGORY: Compile Manager
SCOPE: APPLICATION
CLASSES: None
PARENT: None

DESCRIPTION:

*** REVISED MAY 2017 ***

This template is used by the CHT Compile Manager and other CHT templates to attach the AACHTControlPanel template to your application where required by CHT templates to provide application configuration services.

CATEGORY:
Dates And Calendars

*** HANDYPOPUPCALENDARBUTTON ***

VERSION: 22A.00.00
TYPE: CONTROL
CATEGORY: Dates And Calendars
SCOPE: WINDOW
CLASSES: HNDDates
PARENT: Window(ABC)

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This code template incorporates the HNDDates Classes and drops code for a popup calendar in the current embed position. The calendar requires 11 standard icons, included with your Clarion Handy Tools install set. These icons should be visible on the path established in Clarion's redirection (.RED) file. The template adds the required icons to the project list where you can see them, in case you've misplaced them for any reason. Example application: HNDCAL.APP

CATEGORY:
Dot NET - FTP

*** EMBEDNETFTPCLASSES ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Dot NET - FTP
SCOPE:	ABC PROCEDURE
CLASSES:	HNDNETFTPClass
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED JUNE 2017 ***

This extension creates a single, local instance of HNDNetFTPClass. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion. Functionality provided is extended Date/Time via a call into a C# DLL built with Visual Studio .NET utilizing COM Interop. The interface layer to Clarion can be found in HNDNETFTP.INC/CLW/TLB.

See HNDNETFTP.INC, HNDNETFTP.CLW and HNDNETFTP.TLB to study the header, code module and method interfaces respectively. Two demonstration applications called HNDFTPNET.APP and HNDZIPNFTPNET.APP illustrate use of the template and calling HNDNetFTPClass methods.

CATEGORY:
EXE Compression

*** HNDCMPDECOUPLEEXECUTABLECOMPRESSOR ***

VERSION:	22A.00.00
TYPE:	UTILITY
CATEGORY:	EXE Compression
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED MAY 2017 *** This template is used by CHT Compile Manager to disable the CHT global templates HandyExecutableCompressor and HandyCodeSigningExtension. This template is executed unconditionally by CHT compile manager to prevent the compile manager from being forced to wait while the compressor batch file is executed or code-signing is performed.

*** HANDYEXECUTABLECOMPRESSOR ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	EXE Compression
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This global extension template compresses your executables (EXEs or DLLs) on completion of compile. This makes your executables smaller by two thirds and lowers web-transport costs without affecting run-time performance. There is also the added benefit of making your executable impervious to password theft and modification with a hex editor, as compression makes all string storage areas of your application completely unreadable. A switch on the template lets you turn this feature off. However, please consult the application interface for complete instructions on how to fully remove or disable this feature once enabled.

HandyExecutableCompressor is a child of global template AACHTControlPanel and as such, must be installed to the global extensions area by first installing AACHTControlPanel and inserting with that template selected. NOT ABOUT ORDER OF OPERATION WHEN CODE SIGNING Since code-signing your executable must take place AFTER other executable file compression this template now detects when the CHT HandyCodeSigningExtension Template is present and enabled and ensures that the signing operation is performed AFTER compression.

CATEGORY:
Email Template

*** EMBEDSMTPFUNCTIONS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Email Template
SCOPE:	ABC PROCEDURE
CLASSES:	HNDSmtp
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

This extension creates a single, local instance of the HNDSmtp Classes. As of Build 19C.00.00 (3rd quarter 2015) this class (HNDSMTP) uses Clarion DLL CLARUNEXT.DLL to perform smtp sends rather than the HNDSM.DLL used in earlier versions. Certain class properties and functions have been dropped or changed and will require amendments if hand-embedded. Template-based calls generated by this template have been adjusted to these class revisions and will generate correctly. Template configurations made on the template prompts may need to be checked for correctness of content but WILL generate correct code for the implementation. Since SMTP is sendmail capable only, the email configurations dialogs have now been adjusted to ask only for SMTP Server, SMTP Port, SMTP User and SMTP Password. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion.

Example applications: HNDZIPNEMAILSMTP.APP, HNDMAILSMTP.APP, HNDBULKSMTPMAIL.APP

*** SMTP CONFIGURE ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	Email Template
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** REVISED DEC 2017 *** This procedure lets the user configure the SMTP settings to the registry such that all portions of your app or system can recover them from there. Important elements of this, like the password are stored encrypted.

*** SMTP SEND HTML ADVANCED ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	Email Template
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** REVISED DEC 2017 *** This procedure provides SMTP send capabilities, using advanced SMTP settings. This function uses SMTP authorization and displays an optional SMTP settings setup interface. HTML can be selected in the form of an HTML file or inserted via code embedded into the procedure. Implements CC, BCC, compression and encryption, embedded images and/or web-referencing images.

*** SMTP SEND TEXT ADVANCED ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	Email Template
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** REVISED DEC 2017 *** This procedure provides SMTP send capabilities, using advanced SMTP settings. It uses SMTP authorization and displays an optional SMTP settings setup interface. Message text can be selected in the form of a file or inserted via code embedded into the procedure. Implements CC, BCC, compress and encryption.

*** SMTP SEND WITH ATTACHMENT COMPRESSION ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	Email Template
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** REVISED DEC 2017 *** This procedure provides SMTP send capabilities, using basic SMTP defaults. Send parameters Smtp Server, To Address, From Address, Subject, Body and Attachment Queue to this procedure or embed constants or variables for these values directly into the procedure.

*** SMTP SEND WITH ATTACHMENTS ***

VERSION:	22A.00.00
TYPE:	JUMPSTART
CATEGORY:	Email Template
SCOPE:	PROCEDURE
CLASSES:	Various
PARENT:	Various

DESCRIPTION:

*** REVISED DEC 2017 *** This procedure provides SMTP send capabilities, using basic SMTP defaults. Send parameters Smtp Server, To Address, From Address, Subject, Body and Attachment to this procedure or embed constants or variables for these values directly into the procedure.

VERSION: 22A.00.00
TYPE: JUMPSTART
CATEGORY: Email Template
SCOPE: PROCEDURE
CLASSES: Various
PARENT: Various

DESCRIPTION:

*** REVISED DEC 2017 *** This procedure provides SMTP send capabilities, using basic SMTP defaults. Send parameters Smtp Server, To Address, From Address, Subject and Body to this procedure or embed constants or variables for these values directly into the procedure.

CATEGORY:
Extended Controls

VERSION: 22A.00.00
TYPE: EXTENSION
CATEGORY: Extended Controls
SCOPE: ABC PROCEDURE
CLASSES: HNDCalc
PARENT: Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This extension creates a single, local instance of the HNDCalcControl Class. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion.

On closing, the calculator posts EVENT:HandyCalcThreadReturn to your calling window at which time the calculator return value can be picked up from module variable CalcInVal. A child control template of this extension template is available (HandyCalcEntryControl_ECF) to assist in calling HandyCalc() from a button that completes an entry field. For a demonstration application, see HNDCALCULATOR.APP.

VERSION: 22A.00.00
TYPE: CONTROL
CATEGORY: Extended Controls
SCOPE: WINDOW
CLASSES: HNDCalcControl
PARENT: Window(ABC)

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This control template places a numeric entry control on your application window along with a button that pops up a CHT HandyCalc multipurpose calculator. The entry field value is passed into the calculator, where it may be manipulated via calculator functions. The resulting calculation can be accepted, in which case the entry control is updated, or rejected, leaving your entry control intact. This template is a child of EmbedCalculatorFunctions (hence the _ECF suffix in the name). That means, EmbedCalculatorFunctions must first be populated before this template becomes visible as a control that may be populated on your

procedure windows. The control brings with it a DECIMAL variable which is automatically populated to the control via PROP:Use. You can, of course change this by selecting a new variable from the DATA button or from your DICTIONARY. The field picture may also be re-assigned as needed. A thread option on this template allows the developer to start the calculator on a dedicated thread that services only the control populated by this template. This feature lets you attach unique instances of the HNDCalcControl class to multiple entry controls on the procedure window without causing cross-contamination from one calculator value to the next or one entry control to the next.

*** HANDYCALCTAPEENTRYCONTROL_ECF ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Extended Controls
SCOPE:	WINDOW
CLASSES:	HNDCalcControl
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This control template places a numeric entry control on your application window along with a button that pops up a CHT HandyCalc multipurpose tape-style calculator. The entry field value is passed into the calculator, where it may be manipulated via calculator functions. The resulting calculation can be accepted, in which case the entry control is updated, or rejected, leaving your entry control intact. This template is a child of EmbedCalculatorFunctions (hence the _ECF suffix in the name). That means, EmbedCalculatorFunctions must first be populated before this template becomes visible as a control that may be populated on your procedure windows. The control brings with it a DECIMAL variable which is automatically populated to the control via PROP:Use. You can, of course change this by selecting a new variable from the DATA button or from your DICTIONARY. The field picture may also be re-assigned as needed. A thread option on this template allows the developer to start the calculator on a dedicated thread that services only the control populated by this template. This feature lets you attach unique instances of the HNDCalcControl class to multiple entry controls on the procedure window without causing cross-contamination from one calculator value to the next or one entry control to the next.

*** HANDYPOPUPMENUBUTTON ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Extended Controls
SCOPE:	WINDOW
CLASSES:	None
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED JUNE 2017 ***

This control template lets you easily create popup menus connected to a button dropped on your application window by the template. For even greater design flexibility, hide the button created with this template and post EVENT:Accepted from another control. Menu execution code may be provided in one of three ways: 1) selecting an executable, 2) entering embed code on the template or 3) entering embed code, inline inside the embed points provided. A CHT Demo application called HandyPopupMenuButton illustrates 5 instances of this template populated on the same window.

Update June 2017: Popup menu configuration now includes full font, size, boldness, color control and programmatic enable/disable capability.

*** HANDYTELLMECONTROL ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Extended Controls

SCOPE:	WINDOW
CLASSES:	HNDWindow
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED FEBRUARY 2017 ***

This control template places a standard text control on your application window. It also creates an embeddable instance of the HNDWindow Classes. The control serves as a message or help control. On the other template buttons, complete the text that you want to initially display.

*** HANDYTOOLBARGRADIENT ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Extended Controls
SCOPE:	WINDOW
CLASSES:	HNDWindow
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

This control template places gradient properties on your toolbar and sizes itself to the toolbar to act as a backdrop. While example TYPE and COLOR settings are assumed by the template, these are, of course, under developer control. The template does assume that the window toolbar is given an equate label, for which the template will prompt you and with which the toolbar can be addressed in code statements made by the template. NOTE: Added Default Color Switch Nov 2017.

*** HANDYWINDOWGRADIENT ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Extended Controls
SCOPE:	WINDOW
CLASSES:	HNDWindow
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

This control template places gradient properties on your window and sizes itself to the window to act as a backdrop. While example TYPE and COLOR settings are assumed by the template, these are, of course, under developer control.

*** RESIZEHANDLESCONTROL ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Extended Controls
SCOPE:	ABC PROCEDURE
CLASSES:	HNDSizes
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED MARCH 2017 ***

This extension template attaches "grip-style" window handles to the bottom right hand corner of your window. Other window controls may be optionally height and/or width synched as the window handle is pulled to stretch or shrink the window. This template creates an instance of the HNDSizes class. As of Feb 2013 it is possible to restrict downsizing of either width or height dimesions of a window below its design-time dimensions in addition to the outright restriction of the entire height or width dimension. This "qualifier" is an extra checkbox for each dimension which may be optionally checked after the fact on any presently full-width, or full-height restricted window. NOTE: This template is identical to the extension template "ResizeHandles" except this version is a "Control" template which can be populated from the window designer and and may be placed automatically on the window by a procedure template.

*** SNAPSENDSTARTBUTTON ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Extended Controls
SCOPE:	WINDOW
CLASSES:	None
PARENT:	ProcedureSnapIns , Window(ABC)

DESCRIPTION:

*** REVISED JUNE 2017 ***

This control template places a single button on your window that may be used to start or stop the CHT SnapSend Utility. This template is a child of CHT template ProcedureSnapIns(HndTools) which must first be available on the procedure before this extended control becomes visible for selection and placement on your procedure window. The ProcedureSnapIns(HndTools) template is itself an auto-populated template that appears automaticallly on any application window procedure when a global template called ApplicationSnapins(HndTools) is added to your application and is configured with its "Enable?" switch checked. This template was added March 2016 for name compatiblity with other SnapxxxStartButton templates (e.g. SnapGetStartButton). It is identical to the template created earlier (which continues to exist) called HNDWINDOWSTOPSTARTSNAPSENDBUTTON.

*** VISITBROWSECOLUMN_EWF ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Extended Controls
SCOPE:	WINDOW
CLASSES:	HNDWindow
PARENT:	EmbedWindowFunctions

DESCRIPTION:

*** REVISED MAY 2017 *** This control enables a weblink in one of your browse columns. The browse may be any ABC browse or any CHT-template-extended browse. It requires that the EmbedWindowFunctions template be placed on the window first. Clicking the browse column enabled by this template starts your default Internet browser - when a valid URL is found in the row being clicked - enabling you to jump to the web site found there. It shares an embeddable instance of the HNDWindow Classes.

This template is a child template of CHT EmbedWindowFunctions and as such, that template must be present before this one can be added.

*** VISITMECONTROL_EWF ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Extended Controls
SCOPE:	WINDOW
CLASSES:	HNDWindow
PARENT:	EmbedWindowFunctions

DESCRIPTION:

*** REVISED OCTOBER 2017 ***

This control template mimics a web hyperlink on your application window. It requires that the EmbedWindowFunctions template be placed on the window first. Clicking the link placed by this template starts your default Internet browser, enabling you to jump to the web site of your choice. It shares an embeddable instance of the HNDWindow Classes. This template is a child template of CHT EmbedWindowFunctions and as such, that template must be present before this one can be added. Click the hyperlink control, to shell to your Internet Browser and visit a website or launch a local HTML, PDF or HELP document of your choice.

CATEGORY:
File And Memory Compression

*** HANDYCOMZIPPERCONTROL ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	File And Memory Compression
SCOPE:	WINDOW
CLASSES:	HNDComZipClass
PARENT:	Window(ABC)

DESCRIPTION:

*** REVISED NOVEMBER 2017 ***

This control template places a CHT COM Zipper Control from HNDZIPR.OCX on your application window. This template provides two interface controls, a string control for messages and a progress control to indicate progress through bulk zip and unzip operations. Since this is a "Control" template it operates from a window. If a silent, or invisible zip operation is required, simply hide the window. Operations provided are: bulk directory zip with mask, bulk directory unzip with mask, single file zip, and single file unzip. The template incorporates a CHT COM Zipper Wrapper called HNDComZipClass into your procedure. See HNDZIPPR.INC, HNDZIPPR.CLW and HNDZIPPR.TLB to study the header, code module and method interfaces respectively. A demonstration application called HNDCOMZIPPER.APP illustrates use of the template and calling HNDComZipClass Class methods.

CATEGORY:
Installation Functions

*** GENERATEHNDSETUPMKRINSTALLDETAILS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Installation Functions
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED NOVEMBER 2017 ***

This extension template provides input screens that allow the developer to create values generated into a web installer shell application like HNDSETUPMKR.APP. At time of writing this template is specific only to be used on HNDSETUPMKR.APP or a copy of that app as it addresses screen controls populated on that app assuming they're there. In the future, if necessary we will expand this particular template with a parent control template that populates the necessary screen controls on a generic window.

For the express purpose of building as many installers as the developer needs, another template of that sort is really not necessary. Developers can simply make a copy of HNDSETUPMKR.APP, customize the

icons, and images and adjust these template prompts as required and have a completed, secure, web-installer in minutes.

The functionality in this template replaces the web-installer generator component that was previously part part of HNDSETUP.APP. Now with HNDSETUP.APP (to create HZO's) and HNDSETUPMKR.APP or a renamed copy of HNDSETUPMKR.APP it is possible to build and compile a complete web-install application. As of build 20C.00, developers are able to also install compiled Clarion Projects as well as compiled Clarion Applications. It is also possible to distribute a ready-to-compile Clarion application or Clarion project in source code.

CATEGORY:
Internet - HTTP Web Servers

*** BROWSERSERVERINITIALIZATIONCONTROLS ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Internet - HTTP Web Servers
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	EmbedBrowserServer

DESCRIPTION:

*** REVISED NOVEMBER 2017 ***

This template creates a set of initialization controls to accompany the EmbedBrowserServer template. These controls collect the information necessary to get a CHT Server application (Browser-Server Or Client-Server) up and running. Information collected includes, server IP, default home directory, server PORT and more. On this template too, you will decide the complexity of your server, whether it is a CHT Static Page Server, a CHT Browser Data Server for dynamic data pages or a CHT Client Server for remote thin-client services. Also determined here is the shape of your server client data base. This aspect applies to data servers only where data security is required. This template reacts to the server type settings enabled on its "Server Configuration" tab so that prompts not relevant to the server type being built are disabled or made invisible.

*** BROWSERSERVERJDOBUILDER ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Internet - HTTP Web Servers
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	HandyJDOAdapter

DESCRIPTION:

*** REVISED MAY 2017 ***

Developer Note: JDO = (J)avascript (D)ata (O)bjects

This template extension works with the HandyJDODataAdapter template to produce JDO-wrapped data to be returned to a remote web page via a passed-in instance of the CHT Browser Server classes. I.E. (HNDSubscriptionServer xServer),BYTE It produces web browse, update, and preview data for use by your HTML and Javascript Web Scripts produced in CHT Scripter in order to render a variety of data-interactive web applications using a Clarion-style browse/form data paradigm. A Web PDF Reports dialog lets you plug in ordinary Clarion reports configured for PDF output and deliver them back to your web pages. This template provides for optionally recycling update forms for continuous insert as well as optionally recycling the update form to the update preview form.

*** CLIENTSERVERDATABUILDER ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Internet - HTTP Web Servers
SCOPE:	ABC PROCEDURE
CLASSES:	None
PARENT:	HandyJDOAdapter

DESCRIPTION:

*** REVISED MARCH 2017 ***

* This template extension works with the HandyJDODataAdapter template to produce client-oriented data streams and packages to be returned to a remote web client via a passed-in instance of the CHT Browser Server classes. It produces client-oriented browse, update, process and report data to service data-interactive, web-client applications using a Clarion-style browse/form data paradigm for lightning-fast, view-oriented, web data clients built with Clarion and CHT templates.

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Internet - HTTP Web Servers
SCOPE:	ABC PROCEDURE
CLASSES:	HNDSubscriptionServer HNDBrowserServer
PARENT:	EmbedBrowserServer

DESCRIPTION:

*** REVISED SEPT 2017 ***

This extension template helps you write code to intercept and process a UCR$ command in your CHT server application, sent to it by a remote client application. A UCR$, or User Custom Request, is any non-standard request dreamed up by the developer that can be performed by a server application on behalf of a requesting client. UCR$ requests, once verified to be from a legitimate, logged-in client, are passed into a server method called "ProcessUserCustomRequest" where the developer can embed code to intercept the request and act on it. This template assists with the "interception" and "acting-on" aspect of that process. UCR$ requests may be accompanied by one or more parameters, so that they emulate remote procedure calls. They can also return single or multiple values, web pages, links, raw data, almost anything the developer requires. User Custom Requests can be run on a thread so that they operate off-line and return immediately, requiring a second call to pick up any data produced, or in-line so that any information returned is passed back immediately on completion of the request. This template is a child of CHT template EmbedBrowserServer and can only be applied to your procedure if that parent template has already been applied.

CATEGORY:
Internet - Sockets

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Internet - Sockets
SCOPE:	ABC PROCEDURE
CLASSES:	HNDSocketEvents
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED JUNE 2017 ***

This extension creates a single, local instance of the HNDSocketEvents Class. All methods appear in the

"Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion. Example application: HNDCHATCLNT.APP, HNDCHATSVR.APP.

CATEGORY:
Querying, Locating And Filtering

*** EMBEDZIPQUERYPARSERFUNCTIONS ***

VERSION: 22A.00.00
TYPE: EXTENSION
CATEGORY: Querying, Locating And Filtering
SCOPE: ABC PROCEDURE
CLASSES: HNDQueryBuilder HNDParse HNDComZipClass HNDDisk HNDUtil HNDError
PARENT: HandyCOMZipperControl

DESCRIPTION:

*** REVISED NOVEMBER 2017 ***

This extension creates a single, local instance of the HNDQueryBuilder and hooks it into a CHT Zip procedure created by the HandyCOMZipperControl template. It allows for CHT-Style queries to filter file selection during the creation of ZIP files. This template is a child of HandyComZipperControl and can only be applied to your procedure if that template is already present.

*** INQUEUEQUERYCONTROL_LBX ***

VERSION: 22A.00.00
TYPE: CONTROL
CATEGORY: Querying, Locating And Filtering
SCOPE: WINDOW PROCEDURE
CLASSES: None
PARENT: ListBoxBrowseExtender

DESCRIPTION:

*** REVISED JUNE 2017 ***

The _LBX suffix on this template name indicates that it is a child template associated only with (L)istbox(B)rowseE(X)tender. This control template places an ENTRY Control *(into which is entered a CHT-QUERY-LANGUAGE value)* and a Clear Query BUTTON Control on your ListBoxBrowseExtender procedure. InQueueQueryControl_LBX appears only in your list of available control templates when control template, ListBoxBrowseExtender is first added to your window. This query template performs a variety of filtering operations like CONTAINS, STARTSWITH, ENDSWITH, EQUALS, and so on, to limit your LBX browse to only those records that have matching values in the fields referenced in the query string. *InQueueQueryControl_LBX* works both in LOCAL DATA MODE using local data tables and Clarion file drivers, as well as in REMOTE DATA MODE via CHT Client Server Connection(s).

Browse refresh data, whichever operating mode is in play, originates from data already in the "MAIN" data queue, filtered data is displayed, while the filter is present, from a temporary "SHADOW" queue. In other words interacting with this field-specific filter, causes no local table or remote client server table access or any sort. Filtering takes place entirely locally, and operates on broader queue data already present in the main data queue.

*** POPFAVORITEQUERIES_CSBL ***

VERSION: 22A.00.00
TYPE: CONTROL
CATEGORY: Querying, Locating And Filtering
SCOPE: WINDOW
CLASSES: HNDBrwFilter HNDClnt

PARENT:	ClientServerBrowseListBox

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

This template creates a set of controls that pop up pre-built queries and query macros saved to a file defined in your dictionary and inserts them into a query entry control also provided by the template. Dropping this template has the combined effect of dropping QueryParsingFilterControl with some features turned off plus PopFavoriteQueries_QPFC and it does so in a single step rather than having to drop the other two templates one after the other. See the predefined tables "PreQuery" and "PreMacro" in HNDACCES.DCT, HNDO.DCT and HNDCLIENTSVLEAN1VIEW.DCT. These tables should be imported into your app's dictionary as a model for pre-defined queries or macros data tables. The table names and type (ISAM or SQL) doesn't really matter to this template, nor do the file prefixes. However, field names and data size definitions are required to be as defined in the PreQuery and PreMacro table designs. CHT has provided an importable JumpStart procedure for browsing, editing and inserting pre-built queries. See the CHT Jumpstart Procedures interface. Queries entered on the query entry control can, of course, be saved into your pre-defined queries table on the fly by users. Deletes of pre-defined queries is also allowed -- no Macro saving or deleting, though. Adding/deleting controls are provided for queries by the template but these may be hidden if you want to disallow all operations except query and macro selection and execution. Demo apps to look at: HNDCLIENTSVLEAN1VIEW.APP, HNDCLIENTSVLEAN4VIEW.APP.

*** POPFAVORITEQUERIES_LBX ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Querying, Locating And Filtering
SCOPE:	WINDOW
CLASSES:	HNDBrwFilter HNDClnt
PARENT:	ListBoxBrowseExtender

DESCRIPTION:

*** NEW DECEMBER 2017 ***

This template creates a set of controls that pop up pre-built queries and query macros saved to a file defined in your dictionary and inserts them into a query entry control also provided by the template. Dropping this template has the combined effect of dropping QueryParsingFilterControl with some features turned off plus PopFavoriteQueries_QPFC and it does so in a single step rather than having to drop the other two templates one after the other. See the predefined tables "PreQuery" and "PreMacro" in HNDACCES.DCT, HNDO.DCT and HNDLBXDM.DCT. These tables should be imported into your app's dictionary as a model for pre-defined queries or macros data tables. The table names and type (ISAM or SQL) doesn't really matter to this template, nor do the file prefixes. However, field names and data size definitions are required to be as defined in the PreQuery and PreMacro table designs. CHT has provided an importable JumpStart procedure for browsing, editing and inserting pre-built queries. See the CHT Jumpstart Procedures interface. Queries entered on the query entry control can, of course, be saved into your pre-defined queries table on the fly by users. Deletes of pre-defined queries is also allowed -- no Macro saving or deleting, though. Adding/deleting controls are provided for queries by the template but these may be hidden if you want to disallow all operations except query and macro selection and execution. Demo apps to look at: HNDLBXDM.APP and HNDPEOPLE_LBX.APP

*** QUERYPARSINGFILTERCONTROL ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Querying, Locating And Filtering
SCOPE:	WINDOW
CLASSES:	HNDBrwFilter
PARENT:	BrowseBox(ABC), Report(ABC), Process(ABC), ExplorerBrowse, HandyMarkerBrowse, ListBoxBrowseExtender

DESCRIPTION:

*** REVISED JUNE 2017 ***

This template creates a query control capable of parsing a programmer-determined query language which can be used to filter data sets. Press the "Query keywords" button to modify the query language or query short forms. When populated on any ABC browse or list box, this template can place the appropriate code to force a filtered browse refresh when a query is entered or cleared. Additional configuration is required depending on which other Clarion Handy Tools templates (if any) you have in place on your browse. This template can also be used in conjuction with the ABC report template to produce filtered reports. The most recent revision allows for more flexibility in control naming when this template is used with ListBoxBrowseExtender. The January 2013 revision allows an option for query keywords to come directly from the translation file rather than being generated into the app from the KEYWORDS dialog. There is also now an option to include a number of internal macros representing the days of the week and the months of the year. By default the macros option is off and the query keywords from translation file is off. The reason for these "Off" defaults is for backward compatibility since these options did not exist prior to January 2013. Unless the developer turns these "On" his app behaves as always.

*** SORTORDERFIELDFILTER_LBX ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Querying, Locating And Filtering
SCOPE:	WINDOW PROCEDURE
CLASSES:	None
PARENT:	[ListBoxBrowseExtender](#)

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

The _LBX suffix on this template name indicates that it is a child template associated only with (L)istbox(B)rowseE(X)tender. This control template places an ENTRY Control *(into which is entered a filter value)* and a Clear Filter BUTTON Control, onto your ListBoxBrowseExtender procedure window. *SortOrderFieldFilter_LBX* appears only in your list of available control templates when control template, *ListBoxBrowseExtender* is first added to your window. This query template performs a CONTAINS query (filter) using INSTRING() to limit your LBX browse to only those records that CONTAIN a matching value in the current sort-order field active on your LBX browse. SortOrderFieldFilter_LBX works both in LOCAL DATA MODE using local data tables and Clarion file drivers, as well as in REMOTE DATA MODE via CHT Client Server Connection(s). Browse refresh data originatesw, of the operating mode is in play, directly from the data table(s). In other , interacting with this field-specific filter, causes the browse to re-access either the file drivers for new data or the remote Client Server for new data, depending on the operating mode in play.

*** UNIVERSALQUERYCONTROL_LBX ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Querying, Locating And Filtering
SCOPE:	WINDOW PROCEDURE
CLASSES:	None
PARENT:	[ListBoxBrowseExtender](#)

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

The _LBX suffix on this template name indicates that it is a child template associated only with (L)istbox(B)rowseE(X)tender. This control template places three controls, ENTRY Control *(into which is entered a CHT-QUERY-LANGUAGE value)*, a Clear Query BUTTON Control and a MAIN or SHADOW QUEUE toggle button, on your ListBoxBrowseExtender procedure.

The UniversalQueryContro_LBX template name derives from the fact that this control issues a query that fills and displays either the MAIN browse queue (from DATA SOURCE, local or remote) or it fills and displays the SHADOW browse queue, by copying from matching records in the MAIN QUEUE. UniversalQueryControl_LBX appears only in your list of available control templates when control template, ListBoxBrowseExtender is first added to your window.

This query template performs a variety of filtering operations like CONTAINS, STARTSWITH, ENDSWITH,

EQUALS, and so on, to limit your LBX browse to only those records that have matching values in the fields referenced in the query string.

UniversalQueryControl_LBX works both in LOCAL DATA MODE using local data tables and Clarion file drivers, as well as in REMOTE DATA MODE via CHT Client Server Connection(s). When displaying MAIN QUEUE DATA a query on this control refreshes from DATA SOURCE. When displaying SHADOW QUEUE DATA this control refreshes from the MAIN QUEUE without interacting at all with the back end DATA SOURCE.

CATEGORY:
SMS Template

*** EMBEDSMSFUNCTIONS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	SMS Template
SCOPE:	ABC PROCEDURE
CLASSES:	HNDSMS
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED JUNE 2017 ***

This extension creates a single, local instance of the HNDSmtp Classes. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion.

Example applications: HNDSMSMESSAGE.APP

CATEGORY:
TXA Handling

*** AAEXPORTPROCEDURE ***

VERSION:	22A.00.00
TYPE:	UTILITY
CATEGORY:	TXA Handling
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED JUNE 2017 ***

This Utility template exports the named procedure into a .TXA. Once export of the procedure is completed, the .TXA is then processed to remove auto-populated procedure templates ProcedureImagesEx, ProcedureSnapIns and ProcedureTransparentControls, if they are present. Leaving auto-populated templates of this sort in the .TXA causes duplication and orphaning of the auto-populated template, since there's no mechanism in the Clarion IDE to re-establish a connection with the ProcedureImagesEx instance being imported or the ProcedureSnapIns instance being imported with the parent template instance that performs the procedure-level population of these templates, namely, ApplicationImagesEx , ApplicationSnapIns, ProcedureTransparentControls templates respectively.

*** AAIMPORTPROCEDURE ***

VERSION:	22A.00.00

TYPE:	UTILITY
CATEGORY:	TXA Handling
SCOPE:	APPLICATION
CLASSES:	None
PARENT:	None

DESCRIPTION:

*** REVISED JUNE 2017 ***

This Utility template imports the named procedure into a .TXA. This template assumes you have first exported the procedure being imported using CHT Utility Template *AAExportProcedure*. Once import of the procedure is completed, the application will be stripped of any duplicate instances of the following CHT auto-populated templates: ProcedureImagesEx, ProcedureSnapIns and ProcedureTransparentControls.

CATEGORY:
Update Controls

*** CHTMULTITABLEUPDATE ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Update Controls
SCOPE:	WINDOW PROCEDURE
CLASSES:	None
PARENT:	Update(ABC)

DESCRIPTION:

*** REVISED JUNE 2017 ***

This control template drops a hidden placeholder control onto your ABC update form procedure window. It writes code to perform an ACCESS:FileName.Update() for every table on your form procedure added to the "Other Files" area of the file tree. When used with a CHT ListBoxBrowseExtender browse using the FillFromView() functionality, this allows you to populate fields on your form that are not necessarily in a single table, but in multiple tables accessed by your LBX View design. Other than positioning from the IDE window editor [optional] no further configurations are necessary on this template interface.

*** UPDATEBUTTONS_LBX ***

VERSION:	22A.00.00
TYPE:	CONTROL
CATEGORY:	Update Controls
SCOPE:	WINDOW PROCEDURE
CLASSES:	None
PARENT:	[ListBoxBrowseExtender](#)

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

The _LBX suffix on this template name indicates that it is a child template associated only with (L)istbox(B)rowseE(X)tender. This control template drops three buttons (Insert, Change and Delete) onto your LBX browse window. Other than positioning and/or cosmetically altering the buttons from the IDE window editor [optional] no further configurations are necessary on this template interface.

CATEGORY:
Utility Functions

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Utility Functions
SCOPE:	APPLICATION
CLASSES:	ABCCWUTILFunctions
PARENT:	AACHTControlPanel

DESCRIPTION:

*** REVISED JUNE 2017 ***

This extension template globally instantiates a single instance of the ABC CWUTIL.INC functions. These functions are not class-based. They appear as extended Clarion language functions everywere in your app if this template has been globally attached to your application. You may change the default instance name if you wish.

This template is a child of AACHTControlPanel and can only be populated to your application once AACHTControlPanel has been added.

CATEGORY:
Window Design And Control

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Window Design And Control
SCOPE:	ABC PROCEDURE
CLASSES:	HNDWindow
PARENT:	AACHTControlPanel ApplicationImagesEx

DESCRIPTION:

*** REVISED DECEMBER 2017 ***

This extension creates a single, local instance of the HNDWindow Classes. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion.

EmbedWindowFunctions manages a host of features on your application window, from application-in-system-tray, to window callbacks, to windowboxes, to dynamic user menus. This template is a child of global template ApplicationImagesEx, which itself is a child of our master global template AACHTControlPanel. These parent-global templates must be populated before this one can be added to any procedure.

As of build 12A1.03 this template also provides a Window Effects dialog that lets you establish degrees of window transparency and to fade a window into view on entry. Various window transitions are also available for bringing windows into view our out of view.

CATEGORY:
Windows API

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Windows API
SCOPE:	ABC PROCEDURE

CLASSES:	HNDFileNotify
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED MAY 2017 ***

This extension creates a single, local instance of the HNDFileNotify Classes. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion. DEMO Application: HNDMONITORDIRECTORYSRVC.APP.

*** SOURCE_EMBEDUTILITYFUNCTIONS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Windows API
SCOPE:	ABC SOURCE PROCEDURES
CLASSES:	HNDUtil
PARENT:	ABCSourceProcedure

DESCRIPTION:

*** REVISED JUNE 2017 ***

This extension creates a single, local instance of the HNDUtil Classes. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion. This template is specific to ABC Source procedures which do not contain the normal embed points found in ABC window procedures like browses, reports and processes.

CATEGORY:
Windows Registry

*** EMBEDREGISTRYFUNCTIONS ***

VERSION:	22A.00.00
TYPE:	EXTENSION
CATEGORY:	Windows Registry
SCOPE:	ABC PROCEDURE
CLASSES:	HNDRegistry
PARENT:	Any ABC/CHT Procedure

DESCRIPTION:

*** REVISED FEBRYART 2017 ***

This extension creates a single, local instance of the HNDRegistry Classes. All methods appear in the "Local Objects" section of the procedure to which this extension is applied and may be embedded in the usual ABC fashion.