AppBuilder
By Magic Software Enterprises

Magic Software AppBuilder

Version 3.2

INI Settings Reference Guide

Corporate Headquarters
Magic Software Enterprises
5 Haplada Street,
Or Yehuda 60218, Israel
Tel +972 3 5389213
Fax +972 3 5389333

INI Settings Reference Guide

Introduction to INI Settings

The *INI Settings Reference Guide* provides the key settings, possible values, default values, and descriptions of initialization files that affect the behavior and operation of the AppBuilder products. Use caution when changing the initialization settings as they may cause unexpected results. An initialization file consists of three parts:

- **Section Names** - Delimited by parentheses () on the mainframe or brackets [ ] in other environments. A Section contains settings for a specific area of the product. For example, [AE Runtime] section of the Hps.ini file contains C runtime settings.
- **Comment Lines** - Begin with a semicolon (;) in position 1.
- **Entry Lines** - Contain the variable name, an equal sign, and the variable's content without quotation marks.

The following illustrates an example section:

```
[AE Runtime]
; This instr_ ucts AppBuilder to search the rule hash directories
; for the rule DLL at runtime.By default, the hash directories
; are located in appbuilder\nt\win\rt\rulebin.
; possible values:
; VAL=TRUE
; VAL=FALSE
RULE_HASH=TRUE
```

We recommend you use the Management Console to view and modify the initialization settings on a workstation wherever possible. For information about how to use the Management Console, refer to the *Communications Guide* .
For the host initialization settings, you can change the variable's settings using the AppBuilder Installation Workbench, or modify them directly using the ISPF utilities.

> ⚠ Initialization settings in this document are listed in the alphabetical order within each section, and differ from the actual order in which they appear in the ini files.

Hps.INI Settings

The Hps.ini file contains initialization information for the operation and configuration of AppBuilder on the workstation.
The Hps.ini file is located directly under the AppBuilder install directory. The settings in the Hps.ini file are relevant to the following products:

- Construction Workbench
- Client Runtime (C Runtime)
- Preparation and Test Server
- Departmental Server
- Gateway Server
- Personal Repository
- Workgroup Repository

Sections in the Hps

The Hps.ini file contains the following sections.

[AE Runtime]	[CONVERTER.CLASSICCOBOL]	[MacroDomains]
[AP DOTNET]	[CServerGen]	[NLS]
[AP GLOBAL]	[CSharpClientGen]	[OpenCobolGen]
[AP Java]	[DBE]	[PERSONAL_REPOSITORY]

[AP Unix]	[FREEWAY]	[REPOSITORIES]
[AP Windows]	[FREEWAY_DLLS]	[RepositoryAdmin]
[APWB]	[FREEWAY_SERVER]	[Repository Client]
[CCommonGen]	[JavaGen]	[REPOSITORY SERVER]
[CobolGen]	[JavaBatchGen]	
[CodeGen]	[JavaHTMLGen]	
[CodegenParameters]	[JavaServerGen]	
[CodegenPragmas]	[JobScheduler]	
[CONVERTER.OPENCOBOL]	[MacroDefinitions]	

{install_dir} in the Possible Value/Default column indicates the AppBuilder installation directory.

AE Runtime

The settings in the [AE Runtime] section are used for the Construction Workbench, C Client Runtime, Preparation & Test Server, and Departmental Server.

Table 2-1 Hps.ini [AE Runtime] section

Key	Possible Values / Default	Description
ALLOW_CHANGE_ PROTECTED_FIELDS	TRUE (default) FALSE	If set to TRUE, it allows fields protected in the Window Painter to have their state changed with the components SET_CONTROL_MODE_BY_ID or SET_FIELD_MODE.
ALWAYS_USE_DNA	YES NO (default)	When set to YES, only the root rule resides locally at runtime. All other rules must reside on the server, and all other rule calls are passed to AppBuilder Communications.
APP_FILE	<valid file name fully qualified by drive and directory> {install_dir}\CTrace.log	This setting instructs AppBuilder where to place the output from "trace" statements generated by C applications.
AUTOSELECT	TRUE FALSE (default)	This setting affects only radio buttons. • TRUE: Navigating between radio buttons moves both the focus and the selection. • FALSE: Navigating between radio buttons only changes focus.
AUTOTAB	ON (default) OFF	When set to ON, when the maximum amount of data is entered into an edit field, focus will move to the next object in the tabbing order. This setting only applies when appending to data in a field. If inserting within existing data, no automatic tab will occur on reaching field limit. When this key is not present or commented out in the INI file, then AUTOTAB=OFF is assumed.
BITMAP_DIR	{install_dir}\NT\RT\BMP	This setting specifies the location of all bitmaps and icon files that are used at runtime or during simulation with the Window Flow Diagram tool. From within the Construction Workbench, this setting must include the [AP Window] section BITMAP_DIR setting. When specifying more than one directory, use a semicolon to separate directories. If this key is not present, \hps\nt\rt\bmp is used as the location of bitmaps and icon files.
CHANGE_PICTURE_FIELDS	YES NO (default)	When set to YES, the setting will left justify picture fields in "compatibility" arithmetic mode for old runtime compatibility.
CHECK_DEC_FORMAT	Y N (default)	This setting controls numeric format string verification. Yes enables verification. No disables verification.

CHECK_MINMAX_ON_ UPDATE_ONLY	TRUE FALSE (default)	This setting controls when min-max range checks are performed for numeric fields defined with min and max range values. When set to FALSE, such range checks are performed on both reading from and updating the window view fields. When set to TRUE, these range checks are performed only when updates are made to the window view fields. Default value is FALSE, preserving the behaviour prior to introduction of this setting.
CLEAR_ALTERED_FIELD_ STATUS_ON_CONVERSE	TRUE FALSE (default)	When set to TRUE, objects that are marked with an altered state will be reset on the next converse.
COMBOBOX_SEL_EVENT	TRUE FALSE (default)	When set to TRUE, a combo box returns an event when a selection is made. The combo box generates an event when the user presses enter, selects with the mouse, moves focus, or directly changes the edit field.
COMMS_HANDLER	httpint	This setting enables the HTTP interface.
CONFIRM_PRINTSCREEN	TRUE (default) FALSE	If set to TRUE, the user will be prompted with a confirmation window when doing a print screen.
CONVERSE_NOWAIT_EVENT	TRUE (default) FALSE	When set to TRUE, using "converse window WINDOW_NAME nowait" in the rules code causes an HPS_NOWAIT event to be generated.
CONVERSE_NOWAIT_ RETURNS_EVENT	TRUE (default) FALSE	This setting specifies whether or not the converse window nowait call removes the events from the system event queue or not.
CTL3D	TRUE FALSE (default)	This setting specifies the default for the Windows 3D effect for windows that do not define this attribute themselves. AppBuilder/HPS Windows that have not been modified in HPS541 NT with FP6n Window Painter and above do not contain this attribute.
CUSTOM_ERROR_TEXT	<text up to 50 characters> Some Fields are in error. (default)	This setting specifies the text for a message box that is displayed when selecting a menu item or a push button, and the window contains fields marked as being in error. Individual menu items and push buttons may override this behaviour by using their 'Ignore validation' setting.
D0_CHECK	YES NO (default)	When set to YES, execution will stop with a system exception if division by zero occurred with dec/pic arithmetic operations.
DEFAULT_CENTURY	1900 (default)	This setting specifies the value for the century in the DATE function when only two digits are used for the year in the input and format strings. This value is added to the two digit year parsed.
DETACH_WINDOW_TO_TOP	TRUE FALSE (default)	When set to TRUE, when a detached rule converses a window, the window is made the foreground window and receives focus. When set to FALSE, the parent window retains focus.
DISABLE_SCREEN_PRINT	YES NO (default)	This setting specifies whether or not to disable the print screen functionality at runtime.
EDITFIELD_BORDER	ON OFF (default)	This setting determines the default border attribute for protected edit fields without this attribute specified. Only windows last edited in 5.2 are lacking this attribute for protected edit fields.
ENABLE_MCLB_COLUMN_HELP	TRUE FALSE (default)	If set to TRUE, when requesting help and focus is in an MCLB, the HPS ID of the column is passed to the help system rather than that of MCLB. This only applies when using the GUI help.
ERROR_SHOW_COLOR	TRUE (default) FALSE	When set to TRUE, the background color of a field changes to red when invalid data is entered.
ERROR_SHOW_STARS	TRUE FALSE (default)	When set to TRUE, the invalid data entered is replaced with 'stars' (asterisks).

FIELD_CHANGED_ON_TYPING	TRUE FALSE (default)	This setting controls how edit fields and MCLB cells are determined to have been modified. When set to FALSE, these controls are determined to have been modified by comparing the control's value against the corresponding field in the window view. When set to TRUE, this determination is made upon whether the user typed into the field, even if overtyping with the same value. When a field is determined to have been modified, any validation such as min-max range checks will be performed. Default value is FALSE, preserving the behaviour prior to introduction of this setting.
FILTER_KEYCODES	<comma separated list of key codes> Default is none.	This setting prevents a character from being entered into an edit field. The characters that are to be filtered must be represented with their key code number. For example, if FILTER_KEYCODES=97,98,99 then the letters a, b, and c would be filtered.
FILTER_KEYCODES_BEEP	TRUE FALSE (default)	When a key in the FILTER_KEYCODES list is entered, a beep will sound if the setting is TRUE.
GET_USERID_FROM_DNA	TRUE FALSE (default)	If set to TRUE, AppBuilder will retrieve the userid from AppBuilder Communications when the system components GET_FULL_USER_IDENTITY and GET_USER_WORKSTATION_ID are called. AppBuilder Communications will invoke the authentication exit specified in its settings to retrieve the userid (and password) if not already done. If set to FALSE, the userid field for these system components will be left blank.
GUI_HELP_DIR	{install_dir}\NT\RT\HLP	This setting specifies the location of the native operating system GUI help file(s). The actual help files are specified using the HPS_SET_HELP_FILE_NAME system component; once a help file has been specified with the system component, access to the AppBuilder's basic help is disabled. For more information about using help, refer to the *Developing Applications Guide* .
HELP_DIR	{install_dir}\NT\RT\HLP	This setting specifies the location of the AppBuilder help files that were created in the Window Painter. This setting must typically match the directory specified in the HELP_DIR setting under the [AP Windows] section if the preparation environment must also be set. For more information about using help, refer to the *Developing Applications Guide* .
HPS524_COMPATIBLE	TRUE FALSE (default)	This setting affects a number of aspects of the AppBuilder runtime in order to simulate behavior of HPS 5.2. It is provided as a migration aide. This setting affects the following functionality: 1. AppBuilder *default behavior* : Disabled edit controls, whether they are disabled in the Window Painter or disabled at runtime using the system components, are displayed with the text grayed out. *HPS 5.2.4 simulated behavior* : Disabled edit controls are not grayed out. 2. AppBuilder *default behavior* : ReadOnly fields are included in the tab order. *HPS 5.2.4 simulated behavior* : ReadOnly fields are not included in the tab order. 3. AppBuilder *default behavior:* On double-clicking a protected MCLB cell with immediate return enabled, the generated event has the physical view occurrence in the EVENT_PARAM field. HPS 5.2.4 simulated behavior: The EVENT_PARAM field contains the virtual view occurrence for the double-clicked cell. 4. AppBuilder *default behavior:* Initial window position is adjusted to ensure that no part is off-screen (subject to the window being smaller than the screen). HPS 5.2.4 simulated behavior: Initial window position is as specified in Window Painter.

HPSRT_DEBUG_START	QUERY (default) TRUE FALSE	This setting specifies whether or not to load RuleView on starting application. • QUERY: A window displays at application startup and asks whether to start RuleView or not. If you choose not to load RuleView at this time, you can still load RuleView by selecting Alt+Pause during execution. • FALSE: Do not load RuleView. ALT+Pause will not work at execution time. • TRUE: The RuleView debugger is always started when the application is run. For more information on debugging, refer to the *Debugging Applications Guide* .
IMMEDIATE_RETURN_ ON_MCLB_ENTER	TRUE (default) FALSE	When set to TRUE, an IMMEDIATE_RETURN_EVENT is generated when the Enter key is pressed when focus is in a cell in an MCLB column marked as protected and immediate return.
INDEX_CONTROL_ABORT	YES NO (default)	This setting specifies whether or not to abort the rule at execution time if index is out of bounds when accessing occurring views. See also INDEX_CONTROL_ON setting in the [CodegenParameters] section.
INIT_DNA_AT_STARTUP	TRUE FALSE (default)	When set to TRUE, AppBuilder Communications is automatically started when runtime is started. AppBuilder Communications initialization includes calling the client authentication exit, if configured, and connecting to any specified database.
KEY_ESC_EVENT	TRUE FALSE (default)	When set to TRUE, pressing the "ESC" key causes an event.
LANGUAGE	ENU1004 (US-English) (default) dan1004 (Danish) deu1004 (German) eng1004 (English) esp1004 (Spanish) fra1004 (French) grk1253 (Greek) ita1004 (Italian) jpn932 (Japanese) kra949 (Korean) ptg1004 (Portuguese) tha874 (Thai)	This setting identifies the codepage to use at runtime. The codepage that is chosen must match the codepage that is set in the [NLS] section. Verify that the NLS section Codepage_WIN setting is the same as the number given in the LANGUAGE setting. This setting also determines the language for messages. When installing in a Japanese environment, it is set to JPN932. When installing in a Korean environment, it is set to KRA949.
LISTBOX_SEL_ORDER	NATURAL (default) SELECTION	When set to NATURAL, the component GET_SELECTED_FIELD returns the selected rows in an MCLB or list box in ascending order. If set to SELECTION, the component returns the rows in the order in which the user selected them.
MCLB_CELL_XPADING	<small integer> 3 (default)	This setting defines the width of a border, in pixels, between cell contents and the MCLB column edges.
MCLB_CHANGE_PROTECTED	TRUE FALSE (default)	If set to TRUE, it is possible to use the component SET_CONTROL_MODE_BY_ID or SET_FIELD_MODE to enable or disable MCLBs or MCLB columns.
MENU_FILE	MENUFILE.MNU (default)	This setting specifies the name of the file that contains the information used to create the menu when executing the 'AppBuilder\Execution Clients\Windows Client' program menu item (AESHELL.EXE). This setting must typically match the setting specified in the MENU_FILE setting under the [AP Windows] section if the preparation environment must also be set.
MENUFILE_DIR	{install_dir}\NT\RT\MENU	This setting specifies the location of the file associated with the MENU_FILE setting. This setting must typically match the directory specified in the MENUFILE_DIR setting under the [AP Windows] section if the preparation environment must also be set.

MESSAGES	Info Warnings Errors Errors+Info (default) Info+Warnings Errors+Warnings Info+Warnings+Errors	This setting specifies the runtime display message types. It can be any combination of the Error, Warnings, and Info. The values are case sensitive.
OVERFLOW_CHECK	YES NO (default)	When set to YES, it makes execution stop with system exception if an overflow occurred with dec/pic arithmetic operations.
PANEL_BUFFERS	<numeric value from 0 to 100> 5 (default)	This setting caches the panel files for quicker performance. Increasing the number increases the amount of memory used.
PANEL_DEFAULT_COLOR	TRUE FALSE (default)	When set to TRUE, AppBuilder panels designed to use default color will use the default color of the system.
PANEL_DIR	{install_dir}\NT\RT\PNL	This setting specifies the location of the AppBuilder PNL files. PNL files are used to create the GUI windows at runtime. This setting must typically match the directory specified in the PANEL_DIR setting under the [AP Windows] section if the preparation environment must also be set. For C runtime in the MLUI enabled environment, you must also specify the language specific directory if the current language is different from the default language. Use a semicolon to separate the entries. For example, if French is specified as the current language, modify this setting as follows: PANEL_DIR={install_dir}\NT\RT\PNL;{install_dir}\NT\RT\PNL\French The "langauge" variable is not case-sensitive. You do not need to make this same change in the REF_DIR setting under the [AP Windows] section when using the MLUI. Refer to the *Multi-Language User Interface Guide* for information about MLUI environment.
PC_BATCH_INI	PCBATCH.INI	This setting specifies the location of the PCBATCH.INI file, which is used by the PC batch components.
REF_DIR	{install_dir}\NT\RT\REF	This setting specifies the location of all the files that contain the information and data for Sets at runtime. This setting must typically match the directory specified in the REF_DIR setting under the [AP Windows] section if the preparation environment must also be set. For C runtime in the MLUI enabled environment, you must also specify the language specific directory if the current language is different from the default language. Use a semicolon to separate the entries. For example, if French is specified as the current language, modify this setting as follows: REF_DIR={install_dir}\NT\RT\REF;{install_dir}\NT\RT\REF\French The "language" variable is not case-sensitive. You do not need to make this same change in the REF_DIR setting under the [AP Windows] section when using the MLUI. Refer to the *Multi-Language User Interface Guide* for information about MLUI environment.
RULE_BUFFERS	<value from 0 to100>	This setting specifies the buffer size to cache the AppBuilder rule DLLs for better performance. The higher the number is set, the more memory will be used.
RULE_CFG_DIR	{install_dir}\NT\RT\PROCESS	This setting specifies the location of the process.lst and configuration files using RuleView debugging. This setting must match the [AP Windows] section RULE_CFG_DIR setting.
RULE_DLL_DIR	{install_dir}\NT\RT\RULEBIN	This setting specifies the location of the AppBuilder rule DLLs used at runtime. When specifying more than one directory, use a semicolon to separate directories. This setting must match the [AP Windows] section RULE_DLL_DIR setting.
RULE_DLL_DIR_SRV	{install_dir}\NT\RT\RULEBIN	This setting specifies the location of the AppBuilder server rule DLLs used at runtime.

RULE_FAIL_SILENT	TRUE FALSE (default)	This setting determines whether or not to display the standard AppBuilder Communications error dialog when a AppBuilder Communications error occurs. The behavior is also dependent on the USE_DNAERROR_RULE setting. • If USE_DNAERROR_RULE=TRUE and RULE_FAIL_SILENT=FALSE, then HPS_COMM_ERROR_RULE takes precedence with invocation of RDNAERR and the standard AppBuilder Communications error dialog is not displayed. • If USE_DNAERROR_RULE=FALSE and RULE_FAIL_SILENT=FALSE, then the standard AppBuilder Communications error dialog is displayed. • If USE_DNAERROR_RULE=TRUE and RULE_FAIL_SILENT=TRUE, then HPS_COMM_ERROR_RULE takes precedence with invocation of RDNAERR and the standard AppBuilder Communications error dialog is not displayed. • If USE_DNAERROR_RULE=FALSE and RULE_FAIL_SILENT=TRUE, then the error is ignored since both HPS_COMM_ERROR_RULE and RDNAERR are suppressed. In all cases, the rule execution continues after the point of failure, unless the code specifies otherwise.
RULE_HASH	TRUE FALSE	This setting specifies whether or not AppBuilder searches the rule hash directories for the rule DLLs at runtime. By default the hash directories are located in appbuilder\nt\win\rt\rulebin. This setting must match the [AP Windows] section RULE_HASH setting.
RULE_SRC_DIR	{install_dir}\NT\RT\RULESRC	This setting specifies the location of the AppBuilder rule source code, which is used by RuleView for debugging. This setting must typically match the directory specified in the RULE_SRC_DIR setting under the [AP Windows] section if the preparation environment must also be set.
RULEVIEW_INIFILE_DIR	{install_dir}\TEMP	This setting specifies the location of the INI file used by the new RuleView. If the file does not exist it will be created the next time RuleView is run.
SAVE_WINDOW_STATE	TRUE FALSE	This setting specifies whether or not the position and the size of windows are saved when they are closed and restored when windows are reopened. This value must be set to true to use the WindowStateManager plug-in.
SEARCH_HELP_BY_TOPIC	TRUE FALSE (default)	When set to TRUE, the HPS ID is used as a topic/context ID to search a native Windows GUI help file.
SHOW_ERROR_MESSAGES	TRUE (default) FALSE	This setting suppresses all AppBuilder runtime messages.
SHOW_MESSAGE_DURING_REMOTE_CALLS	TRUE FALSE (default)	This setting specifies whether or not to display the status message when a call is made to AppBuilder Communications. The message disappears when the control is returned to the application. This is useful when a remote call is time consuming.
SKIP_BLANK_LINES_IN_SET	TRUE (default) FALSE	This setting specifies whether or not to skip blank lines in the Set text when using the system component GET_TEXT_MESSAGE in C client. • TRUE: The component skips blank lines when returning text from the Set. • FALSE: The component returns blank lines from the Set.
TBL_INIT_BUFFERED	TRUE (default) FALSE	When set to TRUE, the HPS_TBL_INIT_SIZE component is BUFFERED. If set to FALSE, the component is IMMEDIATE. If the component is IMMEDIATE, an event must be generated before the MCLB is initialized.

TBL_INIT_EVENT	TRUE (default) FALSE	When set to TRUE, an event is generated if the component HPS_TBL_INIT_SIZE is used. The new MCLB always generates an event when this component is used.
TEMP_SINGLE_SEL_EVENT	TRUE FALSE (default)	When set to TRUE, MCLBs behaves in the following manner: • When a row is selected, an HPS_SELECT event is generated. • When a row is deselected, an HPS_DESELECT event is generated. (For single select MCLBs, a row is deselected by clicking on another row; for multiple selection MCLBs, by clicking on an already selected row.) The HPS_DESELECT event is always generated before the HPS_SELECT event.
TEMPORARY	{install_dir}\NT\TEMP	This setting specifies the temporary directory used by runtime.
THUMB_DRAG	TRUE FALSE	When set to TRUE, the rows of the MCLB scroll when the vertical elevator button is used.
UCOMP_DIR	{install_dir}\NT\RT\CMPBIN	This setting specifies the location of the user component DLLs. This setting must typically match the setting specified in the COMP_DLL_DIR setting under the [AP Windows] section if the preparation environment must also be set.
USE_DNAERROR_RULE	TRUE FALSE (default)	This setting determines whether or not the HPS_COMM_ERROR_RULE is invoked when a AppBuilder Communications error occurs. The behavior is also dependent on the RULE_FAIL_SILENT setting. Refer to RULE_FAIL_SILENT for more information.
USE_OLD_SET_SELECTED_FIELD	TRUE FALSE (default)	This setting determines how the selections are made in a list box or multicolumn list box (MCLB) when using the HPS_SET_SELECT_FIELD system component. Refer to the *System Components Reference Guide* for more information.
USE_SYSTEM_DISABLING_FOR_PROTECTED_FIELDS	TRUE FALSE (default)	When an editable field is disabled/protected in AppBuilder, the text in the field remains black if the setting is FALSE. If set to TRUE, the system handles the coloring of the text.
VALIDATE_DATE	TRUE (default) FALSE	When set to TRUE, the format for dates is checked.
WINDOW_STATE_KEY	VAL=<Key>	This setting specifies the key that is used by Window State Manager. The user specific settings are stored in the registry under HKCU\Software\BluePhoenix\AppBuilder\Runtime\<Key>. Note that <Key> must consist only of alphanumeric single-byte characters.
WINDOW_STATE_MANAGER	{install_dir}\NT\RT\WindowStateManager.dll (default)	This setting specifies a plug-in that is used to save/restore position and size of windows.
WORKSTATION_ID		This setting specifies the number that identifies the workstation.
YEARS_IN_THE_FUTURE	<small integer> 30 (default)	When only the last two digits of a year are entered, the century is determined by checking the internal date of the computer and adding the value of this setting.

AP DOTNET

The settings in [AP DOTNET] section are used for the C# Client Preparation.

Table 2-2 Hps.ini [AP DOTNET] section

Key	Possible Values / Default	Description
BITMAP_DIR	{install_dir}\DOTNET\RT	This setting specifies the directory where all the bitmap and icon files are placed by preparation.

COMP_BIN_DIR	{install_dir}\BND	This setting specifies the directory where the AppBuilder User Component's bind files are placed by preparation.
COMP_DLL_DIR	{install_dir}\DOTNET\RT	This setting specifies the directory where the AppBuilder User Component dll is placed by preparation.
COMP_LISTING_DIR	{install_dir}\TEMP	This setting specifies the directory where the temporary files used by Codegen are placed for User Components.
COMPONENT_PACKAGE	component	This is the name of the package to use for components.
CSHARP_COMP_COMP_OPTS	/target:library	This setting specifies any additional optimization options for the C# compiler.
CSHARP_COMPILER_PATH		This setting specifies the C# Compiler path (the csc.exe file location). Use it if the path is not already in the environment variables path. (sample location: C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727)
CSHARP_INC_DIR		This setting specifies the directory where the additional header files are placed. Use it if the directories are not already in the INCLUDE environment variables path. This setting affects Rules and C# User Components.
CSHARP_LIB_DIR		This setting specifies the directory where additional libraries file is placed. Use it if the directories are not already in the LIB environment variables path. This setting affects AppBuilder Rules and C# User Components.
CSHARP_LIB_NAME		This setting specifies the names list of libraries (.lib) or dynamic link libraries (.DLL) that you use. It affects Rules and C# User Components.
DECIMAL_ARITHMETIC_MODE	CALCULATOR	This setting checks the CodeGen decimal arithmetic mode. COMPATIBILITY and COBOL modes are not supported in C# generation.
GLOBAL_PACKAGE		This is the name of the package to use for all rules.
HPS_COPYDLL_DIR	{install_dir}\DOTNET\SYS\BIN	This setting specifies the directory where the AppBuilder C# Framework dll Bphx.sdf.dll, and Bphx.Sdf.Client.dll, are placed.
HPS_COPYLIB_DIR	{install_dir}\NT\SYS\INC	This setting specifies the directory where the AppBuilder headers file is placed. (%HPSINI%\NT\SYS\INC)
HPS_IMPLIB_DIR	{install_dir}\NT\SYS\LIB	This setting specifies the directory where the AppBuilder libraries file is placed. (%HPSINI%\NT\SYS\LIB)
REF_DIR	{install_dir}\DOTNET\RT	This setting specifies the directory where the AppBuilder prepared set executable is placed by Preparation.
RULE_BIND_DIR	{install_dir}\BND	This setting specifies the directory where the AppBuilder Rule's bind files are placed by preparation.
RULE_COMP_OPTS	/target:winexe	This setting specifies any additional optimization options for the C# compiler. It is used to compile the AppBuilder Rules.
RULE_COMP_SERVER_OPTS	/target:library	This setting specifies any additional optimization options for C# compiler. It is used to compile AppBuilder C# Server Rules.
RULE_DLL_DIR	{install_dir}\DOTNET\RT	This setting specifies the directory where the AppBuilder rule dll file is placed by preparation.
RULE_LISTING_DIR	{install_dir}\TEMP	This setting specifies the directory where the temporary files used by Codegen are placed for Rule prepare.
SET_COMP_OPTS	/target:library	This setting specifies any additional optimization options for the C# compiler. It is used to compile the AppBuilder Sets.
SET_GENERATION	static (default) dynamic	This setting specifies codegen to refer generated sets as static or dynamic.
SET_PACKAGE	set	This is the name of the package to use for sets.
Supported	Y (default) N	This setting enables Windows Application on Project Option setting.
SYSTEM_SET_PACKAGE	Bphx.Sdf.Common.Sets	This is the name of the system set package. Do not modify this file. If you modify this setting, preparation fails.

SYSTEM_VIEW_PACKAGE	Bphx.Sdf.Common.Views	This is the name of the system view package. Do not modify this file. If you modify this setting, preparation fails.
VIEW_COMP_OPTS	/target:library	This setting specifies any additional optimization options for the C# compiler. It is used to compile the AppBuilder Views.
VIEW_PACKAGE	view	This is the name of the package to use for views.

AP GLOBAL

The settings in [AP GLOBAL] section are used for the Construction Workbench and Preparation & Test Server.

Table 2-3 Hps.ini [AP GLOBAL] section

Key	Possible Values / Default	Description
BIND_DB_OPTION	-o <valid bind options>	This setting specifies any additional options for DB2/UDB database Bind for generating C client/server database rules. The value must start with -o followed by a valid bind option. To specify more than one option, separate the options with a space. The example below sets two precompile options: BIND_DB_OPTION=-o DYNAMICRULES RUN
CONVERT_ALL_HOST_VARS	Y N (default)	This setting causes RULEPREP to tell CODEGEN to generate code that re-initializes all of the rule's HPS temp SQL variables with the corresponding host variable values prior to executing the imbedded SQL. This causes the rule to incur additional runtime overhead. Set this to Y only if you write HPS RULEs that modify a "host variable" between two interdependent "sql asis" blocks.
DB_TBL_CASE_OPTION	AS_IS (default) UPPER_CASE LOWER_CASE	This setting specifies whether the name of the table and column to be uppercase, lowercase, or as is.
DBCSEnable	Y (default) N	This setting is set on install. When set to Y, the preparation supports DBCS (double-byte character set). It only appears on a DBCS-enabled computer.
DDL_FMT_DIR	{install_dir}\AD\CFG\DATA	This setting specifies the location of the DDLFMT.dat that specifies the data type used by AppBuilder internally to create DDL file.
DEBUG_DIR	{install_dir}\DBG	This setting specifies the directory where the intermediate files are placed by preparation.
DEC_AS_DB22FLOAT	Y N (default)	When set to Y, decimal value larger than 15 will not be convert to character field when creating DDL generation.
HPSCATSFILES	{install_dir}\MSG	This setting specifies the path to the directory containing the message catalogues.
JEE_TMPL_DIR	{install_dir}\AD\CFG\DATA	This setting specifies the location of the xtpl files directory shipped with AppBuilder.
LANG	C (default)	This setting specifies the location of the Messaging files.
ORACLE_VARCHAR_AS_CHAR	Y N (default)	This setting is used only when using the Oracle database. When set to Y, the VARCHAR fields created with AppBuilder application is converted to the character field when creating AppBuilder runtime dlls.
PREP_DB_OPTION	<valid precompile option>	This setting specifies any additional options for DB2/UDB database precompile for generating C client/server database rules. To specify more than one option, separate the options with a space. The example below sets two precompile options: PREP_DB_OPTION=DYNAMICRULES RUN
PREP_DNA_INI_DIR	{install_dir}\prepclnt	Preparation uses this key to set the DNAINI environment variable.
PREP_SERVER_NAME	ne20 (default)	This setting specifies the name of service using when remote preparation to non host environment.
RES_SERVER_NAME	ne20 (default)	This setting specifies the name of the response service coming back from Remote server.

SKIP_INTEGRITY_CHECK	Y N (default)	When set to Y, when creating menu there is no checking for database rules or components are exist in the hierarchy and will not create menu with database information.
SKIP_MF_RULE_CHECK	Y N (default)	When set to Y, and Include Mainframe Rule is unchecked (in the Construction Workbench menu File > Project Options if a Project is open, or in the Tools > Workbench Option > Preparation tab if a Project is not open), then Mainframe rules' syntax check is skipped. Jobs are still submitted to Prep Status.
TEMPORARY	{install_dir}\TEMP	This setting specifies the temporary directory, which is used by preparation.
VALIDATE_SET	Y (default) N	This setting specifies whether or not to validate a value set of "small integer", "decimal" and "character" types during preparation.

AP Java

The settings in [AP Java] section are used for the Construction Workbench.

Table 2-4 Hps.ini [AP Java] section

Key	Possible Values / Default	Description
ABO_BUILD_ROOT	{install_dir}\JAVA\BLD	This setting specifies the ABO Build directory used for Preparation of ABO Objects.
ABO_ODF_SUB_DIR	odf	This setting specifies the ABO ODF Build sub-directory used to store the ODF files.
BITMAP_DIR	{install_dir}\JAVA\RT\LOCAL\IMAGES	This setting specifies the directory where all bitmap and icon files are placed by the Preparation.
C_COMP_COMP_OPTS	-Zp1 -GB -W3 -MD -LD -DWIN32	This setting specifies any additional optimization options for C compiler.
C_INC_DIR		This setting specifies the directory where additional header files are placed. Use this setting if the directories are not already in the INCLUDE environment variables path. This must be set when preparing C components to be called from a Java application.
C_LIB_DIR		This setting specifies the directory where additional libraries are placed. Use this section if the directories are not already in the LIB environment variables path. This must be set when preparing C components to be called from a Java application.
C_LIB_NAME		This setting specifies the list of names for additional libraries to use. This must be set when preparing C components to be called from a Java application.
COMP_BIND_DIR	{install_dir}\BND	This setting specifies the directory where the AppBuilder C User Component's bind files are placed by preparation.
COMP_LINK_INCS		This setting specifies the additional libraries to link to the C User Components. This must be set when preparing C components to be called from a Java application.
COMP_LINK_OBJS		This setting specifies the additional objects to link to the C User Components. This must be set when preparing C components to be called from a Java application.
COMP_LISTING_DIR	{install_dir}\TEMP	This setting specifies the directory where the temporary files used by the code generation are placed for C User Components.
COMPONENT_PACKAGE	component	This setting specifies the name of the package to use for components.
DB2_VERSION	8	This setting specifies the version of db2profc.exe, DB2 SQLJ Profile Customizer installed on the machine. If this value is not specified, then DB2 Version found in the registry is used. When specified, this value overwrites DB2 version found in the registry. A value of 8 refers to DB2 version 8 and later.

DB_INC		This setting specifies the directory where the database header is placed. Use this setting if the directories are not already in the INCLUDE environment variables path.
DDL_DIR	{install_dir}\JAVA\RT\DBM	This setting specifies the directory where the DDL files are placed by the Preparation.
DECIMAL_ARITHMETIC_MODE	CALCULATOR COBOL COMPATIBILITY	This setting checks the code generation decimal arithmetic mode. See "Decimal Arithmetic Support" section of the *Rules Language Reference Guide* for more information about these arithmetic modes. Three modes are supported: • COBOL: calculator (Windows) • CALCULATOR: default value for Windows • COMPATIBILITY: all other platforms.
GLOBAL_PACKAGE		This setting specifies the name of the package to use for all rules.
HPS_COPYLIB_DIR	{install_dir}\NT\SYS\INC	This setting specifies the directory where the AppBuilder headers are placed.
HPS_IMPLIB_DIR	{install_dir}NT\SYS\LIB	This setting specifies the directory where the AppBuilder libraries are placed.
JAVA_CLASSPATH	{install_dir}\java\RT\appbuilder.jar;.;	This setting specifies the Java classpath. Use this setting if the directories are not already in the CLASSPATH environment variables path.
JAVA_COMP_COMP_OPTS		This setting specifies any additional optimization options for Java compiler.
JAVA_DB2_BIND_OPTIONS		This setting specifies additional db2profc.exe parameters. For DB2 8.1 and later, they are passed as is. prepoptions value: -prepoptions="JAVA_DB2_BIND_OPTIONS"
JAVA_SRC_DIR	{install_dir}\java\RT\LOCAL\JAVASRC	This setting specifies the directory where the AppBuilder server rule JAVA is placed by the preparation.
MENU_FILE	MENUFILE.MNU	This setting specifies the name of the file used for runtime.
MENUFILE_DIR	{install_dir}\java\RT\MENU	This setting specifies the directory where the file associated with the MENU_FILE tag is located.
PACKAGE_LST_DIR	{install_dir}\TEMP	This setting specifies the temporary location of the LST file. This file is created when create package is performed.
RULE_BIND_DIR	{install_dir}\BND	This setting specifies the directory where the AppBuilder Rule's bind files are placed by the Preparation.
RULE_CFG_DIR	{install_dir}\java\RT\LOCAL\DEBUG	This setting specifies the location of the process.lst and configuration files using the RuleView debugging.
RULE_CLASS_DIR	{install_dir}\JAVA\RT\LOCAL	This setting specifies the directory of the class files to be placed by the preparation.
RULE_COMP_OPTS		This setting specifies any additional optimization options for C compiler. This option is used to compile AppBuilder Rules.
RULE_DLL_DIR	{install_dir}\java\RT	This setting specifies the directory where the AppBuilder Server rule Class files are placed by the Preparation.
RULE_LISTING_DIR	{install_dir}\TEMP	This setting specifies the directory where the temporary files used by the code generation are placed for the Rule prepare.
RULE_SRC_DIR	{install_dir}\java\RT\LOCAL\DEBUG	This setting specifies the directory where the AppBuilder rule Sources are placed by the Preparation. These files are used by the RuleView debugging and saved only when the Rule Debug option under the Construction Workbench menu Tools -> Workbench option -> Preparation tab is set.
SET_PACKAGE	set	This setting specifies the name of the package to use for sets.

SYSTEM_SET_PACKAGE	appbuilder.systemviews	This setting specifies the name of the system set package. Do not modify this setting. If you modify it, preparation will fail.
SYSTEM_VIEW_PACKAGE	appbuilder.systemviews	This setting specifies the name of the system view package. Do not modify this setting. If you modify it, preparation will fail.
THINCLIENT_ERROR_HTML	c:\defaulterr.html	This key is commented out by default. For a thin client configuration, the generated partition includes an HTML error page generated from a default template. The default template can be overridden by setting the path to the new template files by setting the values.
THINCLIENT_START_HTML	c:\defaultstart.html	This key is commented out by default. For a thin client configuration, the generated partition includes an HTML start page generated from a default template. The default template can be overridden by setting the path to the new template files by setting the values.
TOP_LABEL	MENU (default)	This setting specifies the display name of runtime menu.
USE_JPUBLISHER	Y N	This setting allows the use of Oracle's JPublisher when preparing database rules and components for Java using Oracle. When set to the default, Y, JPublisher is used for SQLJ pre-processing when preparing for Oracle. When set to N, SQLJ.exe continues to be used.
VIEW_PACKAGE	view	This setting specifies the name of the package to use for views.
WAS_HOME	<drive and directory where WebSphere is installed>	This key is commented out by default. For creating the deployable EJBs archive, set the WebSphere home directory. For example, WAS_HOME= c:\WebSphere\AppServer

AP Unix

The settings in [AP Unix] section are used for the Construction Workbench.

Table 2-5 Hps.ini [AP Unix] section

Key	Possible Values / Default	Description
COMP_LISTING_DIR	{install_dir}\TEMP	This setting specifies the directory where the temporary files used by the Code generation are placed for C User Components.
RULE_LISTING_DIR	{install_dir}\TEMP	This setting specifies the directory where the temporary files used by the Code generation are placed for Rules.
SAVE_WEBSERVICE_FILES_DIR	{install_dir}\WebService\Unix	This setting specifies the directory where the BIND and WSDL files are stored for WebService frontier rules.
SRV_RULE_COMP_OPTS	-DUNIX -DTSERVER	This setting specifies the Server Rules Compiler option.

AP Windows

The settings in [AP Windows] section are used for the Construction Workbench and Preparation & Test Server.

Table 2-6 Hps.ini [AP Windows] section

Key	Possible Values / Default	Description
BITMAP_DIR	{install_dir}\NT\RT\BMP	This setting specifies the directory where all bitmap and icon files are placed by Preparation. This setting must match the [AE Runtime] section BITMAP_DIR setting.
C_COMPILER_PATH		This setting specifies the Compiler path. Use this section if the path is not already in the environment variables path.
C_INC_DIR		This setting specifies where to place additional directories for header files if the directories are not already in the Include environment variables path. This setting affects AppBuilder rules and user components prepared for C.

C_LIB_DIR		This setting specifies where to place additional directories for libraries if the directories are not already in the LIB environment variables path. This setting affects AppBuilder rules and user components prepared for C.
C_LIB_NAME		This setting specifies where to list the names of additional libraries you are using. This setting affects rules and components prepared for C.
COMP_BIND_DIR	{install_dir}\BND	This setting specifies the directory where the AppBuilder C User Component's bind files are placed by preparation.
COMP_COMP_OPTS	-Zp1 -GB -W3 -MD -LD -DWIN32	This setting specifies any additional optimization options for C compiler. This setting is used only for Client install. For Prep Server install, use SRV_COMP_COMP_OPTS.
COMP_DLL_DIR	{install_dir}\NT\RT\CMPBIN	This setting specifies the directory where the AppBuilder C User Component DLL are placed by Preparation. This setting must match the [AE Runtime] section UCOMP_DIR setting.
COMP_INC_DIR	{install_dir}\TEMP	This setting specifies the directory where the AppBuilder Server Rule's headers created by preparation are placed. This setting is used only for Prep Server install.
COMP_LINK_INCS		This setting specifies where to place additional libraries to link to the C User Components.
COMP_LINK_OBJS		This setting specifies where to place additional objects to link to the C User Components. This setting is used only for Client install. For Preparation Server install, use SRV_COMP_LINK_OBJS.
COMP_LINK_OPTS	/DLL	This setting specifies any additional Linker option for the C User Components. This setting is used only for Client install. For Preparation Server install, use SRV_COMP_LINK_OPTS
COMP_LISTING_DIR	{install_dir}\TEMP	This setting specifies the directory where the temporary files used by the Code generation are placed for C User Components.
COMP_RC_FILE	<location\file_name.rc>	This setting specifies the location and name of the Resource files (extension .rc) used by the C User Components.
COMP_RC_OPTS		This setting specifies any additional options for Resource compile. This option is used to compile AppBuilder C User Components.
DB_INC		This setting specifies the directory where the Database headers is placed. Use this section if the directories are not already in the INCLUDE environment variables path.
DDL_DIR	{install_dir}\NT\RT\DBM	This setting specifies the directory where the DDL files are placed by Preparation.
DECIMAL_ARITHMETIC_MODE	CALCULATOR COMPATIBILITY COBOL	This setting checks the code generation decimal arithmetic mode. See "Decimal Arithmetic Support" section of the *Rules Language Reference Guide* for more information about these arithmetic modes. Three modes are supported: • COBOL: calculator (Windows) • CALCULATOR: default value for Windows • COMPATIBILITY: all other platforms.
HELP_DIR	{install_dir}\NT\RT\HLP	This setting specifies the directory where the AppBuilder-generated online help files are placed by Preparation. This setting must match the [AE Runtime] section HELP_DIR setting.
HPS_COPYLIB_DIR	{install_dir}\NT\SYS\INC	This setting specifies the directory where the AppBuilder headers is placed.
HPS_IMPLIB_DIR	{install_dir}\NT\SYS\LIB	This setting specifies the directory where the AppBuilder libraries are placed.
IMPNAME_FILE	{install_dir}\NT\RT\IMPNAME.LST	This setting specifies the path and name of a text file containing lines with two case-sensitive fields: the first field is the name of the requested service, the second is the path and name of the DLL to be accessed in response to a request for the service. The first field in each entry begins in column 1. This setting is used only for the Preparation Server install.

LINKER_PATH		This setting specifies the Linker path. Use this section if the path is not already in the environment variables path.
MENU_FILE	MENUFILE.MNU	This setting specifies the name of the file that contains the information used to create the menu when AESHELL.EXE is run. This setting must match the [AE Runtime] section MENU_FILE setting.
MENUFILE_DIR	{install_dir}\NT\RT\MENU	This setting specifies the location of the file associated with the MENU_FILE tag. This setting must match the [AE Runtime] section MENUFILE_DIR setting.
PACKAGE_LST_DIR	{install_dir}\TEMP	This setting specifies the directory where the packaging files are placed temporary by Preparation.
PANEL_DEBUG	Y N (default)	If this setting is Y, then the extracted repository panel file is saved. Window preparation needs to update saved .REP instead of .PNL, which will be overwritten by runtime .pnl.
PANEL_DIR	{install_dir}\NT\RT\PNL	This setting specifies the directory where the AppBuilder-generated Window files are placed by Preparation. This setting must match the [AE Runtime] section PANEL_DIR setting.
PREP_VERBOSE	Y N (default)	When set to Y, the Verbose information are written to the preparation status. This setting is used only for the Preparation Server install.
RC_PATH		This setting specifies the Resource Compiler path. Use this section if the path is not already in the environment variables path.
REF_DIR	{install_dir}\NT\RT\REF	This setting specifies the directory where the AppBuilder prepared set executable are placed by Preparation. This setting must match the [AE Runtime] section REF_DIR setting.
RULE_BIND_DIR	{install_dir}\BND	This setting specifies the directory where the AppBuilder Rule's bind files are placed by Preparation.
RULE_CFG_DIR	{install_dir}\NT\RT\PROCESS	This setting specifies the location of the process.lst and configuration files using RuleView debugging. This setting must match the [AE Runtime] section RULE_CFG_DIR setting.
RULE_COMP_OPTS	-Zp1 -GB -W3 -MD -LD -DWIN32	This setting specifies any additional optimization options for C compiler. This option is used to compile AppBuilder Rules. This setting is used only for Client install. For the Preparation Server install, use SRV_RULE_COMP_OPTS.
RULE_DLL_DIR	{install_dir}\NT\RT\RULEBIN	This setting specifies the directory where the AppBuilder Server rule DLL are placed by Preparation. This setting is used only for Client install. For the Preparation Server install, use SRV_RULE_DLL_DIR. This setting must match the [AE Runtime] section RULE_DLL_DIR setting.
RULE_HASH	TRUE (default) FALSE	This instructs AppBuilder to copy the rule DLLs to the rule hash directories. By default the hash directories are located in appbuilder\nt\rt\rulebin. This setting must match the [AE Runtime] section RULE_HASH setting.
RULE_INC_DIR	{install_dir}\TEMP	This setting specifies the directory where the AppBuilder Server C User Component's headers created by preparation are placed. This setting is used only for the Preparation Server install.
RULE_LINK_OPTS	/DLL	This setting specifies any additional Linker options for AppBuilder Rules. This setting is used only for Client install. For the Preparation Server install, use SRV_RULE_LINK_OPTS.
RULE_LISTING_DIR	{install_dir}\TEMP	This setting specifies the directory where the temporary files used by Codegen are placed for Rule prepare.
RULE_RC_OPTS		This setting specifies any additional options for Resource compiler. This option is used to compile AppBuilder Rules.
RULE_SRC_DIR	{install_dir}\NT\RT\RULESRC	This setting specifies the directory where the AppBuilder rule Sources are placed by Preparation. These files are used by RuleView debugging and saved only when Rule Debug option under Tools -> Workbench option -> Preparation tab is set. This setting must match the [AE Runtime] section RULE_SRC_DIR setting.

SAVE_TEMP	Y N (default)	When set to Y, the intermediate files are save to DEBUG_DIR. This setting is used only for the Preparation Server install.
SRV_COMP_COMP_OPTS	-W2 -Zp1 -GB -MD -LD -Os -D_*STDC*_ -DWIN32 -DTSERVER -D_*SERV*_ -D_*CONNECTION*_ -DNEWSRVPREP	This setting specifies any additional optimization options for C User Components. This option is used to compile AppBuilder Server C User Components. This setting is used only for the Preparation Server install. For the Client install, use COMP_COMP_OPTS.
SRV_COMP_DLL_DIR	{install_dir}\NT\RT\SERVER	This setting specifies the directory where the AppBuilder Server C User Component DLL are placed by Preparation. This setting is used only for the Preparation Server install. For the Client install, use COMP_DLL_DIR.
SRV_COMP_LINK_OBJS		This setting specifies the names list of additional libraries for Server C User Components you are using. This setting is used only for the Preparation Server install. For the Client install, use COMP_LINK_OBJS.
SRV_COMP_LINK_OPTS	/DLL	This setting specifies any additional Linker options for C User Components. This option is used to link AppBuilder Server C User Components. This setting is used only for the Preparation Server install. For the Client install, use COMP_LINK_OPTS.
SRV_RULE_COMP_OPTS	-W2 -Zp1 -GB -MD -LD -Os -D_*STDC*_ -DWIN32 -DTSERVER -D_*SERV*_ -D_*CONNECTION*_ -DNEWSRVPREP	This setting specifies any additional optimization options for C compiler. This option is used to compile AppBuilder Server Rules. This setting is used only for the Preparation Server install. For the Client install, use RULE_COMP_OPTS.
SRV_RULE_DLL_DIR	{install_dir}\NT\RT\SERVER	This setting specifies the directory where the AppBuilder Server Rule DLL are placed by Preparation. This setting is used only for the Preparation Server install. For the Client install, use RULE_DLL_DIR.
SRV_RULE_HEADER_DIR	{install_dir}\NT\RT\SERVER	This setting specifies the directory where the AppBuilder Server Rule's headers created by preparation are placed. This setting is used only for the Preparation Server install.
SRV_RULE_LINK_OPTS	/DLL	This setting specifies any additional Linker options for AppBuilder Server Rules. This option is used to link AppBuilder Server Rules. This setting is used only for the Preparation Server install. For the Client install, use RULE_LINK_OPTS.
Supported	Y	This setting enables the Windows Application on Project Option setting.
SYS_BIN_SUB_DIR	nt\sys\bin	This setting specifies the location of the runtime execution file Master.exe.
TOP_LABEL	MENU (default)	This setting specifies the display name of runtime menu.

APWB

The settings in [APWB] section are used for the Construction Workbench.

Table 2-7 Hps.ini [APWB] section

Key	Possible Values / Default	Description
APPLICATION_SERVER	WebLogic WebSphere Tomcat WebSphere40	This setting specifies the type of the Web Server. Update this setting from the Construction Workbench menu by choosing *Tools-> Workbench Options-> Preparation* section, Application server drop-down list.
BatchInterfaceDir	{install_dir}\AD\CFG\BIF	This setting specifies the location of the BIF files. If this key is modify, preparation will not work.
DoubleClickOpenPrepDetails	Y N (default)	When set to Y, the Enter key or double click opens the prepare listing instead of opening the source code. You must reopen the Construction Workbench after changing this setting.
PW_BAT_DIR	{install_dir}\AD\CFG\DATA	This setting specifies the directory where the Preparation list is saved when you open for the first time the Construction Workbench.

WP_MULTILINECONTROLS_ALIGNMENT	YES NO	If WP_MULTILINECONTROLS_ALIGNMENT hps.ini key is set to YES, multiline controls will be automatically aligned for compatibility with C runtime. To turn it off, set this key to NO.

The APPLICATION_SERVER setting was moved from [Ap Java] to [APWB] section.

CCommonGen

The settings in [CCommonGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the C generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-8 Hps.ini [CGen] section

Key	Possible Values / Default	Description
DB2DTFMT	%Y-%0m-%0d	This format string is used to convert date fields when calling DB2 Default value is %Y-%0m-%0d.
DB2TMFMT	%0t:%0m:%0s	This format string is used to convert time fields when calling DB2 Default value is %0t:%0m:%0s.
DFLTDTFMT	%0m/%0d/%Y	This format string is used for DATE and CHAR functions when second parameter was omitted. Default value is %0m/%0d/%Y.
DFLTTMFMT	%0t:%0m:%0s	This format string is used for TIME and CHAR functions when second parameter was omitted. Default value is %0t:%0m:%0s.
DFLTTSFMT	%0o-%0d-%Y.%0t.%0m.%0s.%f	This format string is used for TIMESTAMP and CHAR functions when second parameter was omitted. Default value is %0o-%0d-%Y.%0t.%0m.%0s.%0f.
INDEX_CONTROL_ON	YES NO	Specify if generated C code should perform index checking when accessing occurring views. See also INDEX_CONTROL_ABORT in [AE Runtime]

CobolGen

The settings in [CobolGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the ClassicCOBOL generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-9 Hps.ini [CobolGen] section

Key	Possible Values / Default	Description
MACRO	LANGUAGE=Cobol	This setting defines a macro that is language specific.
MACRO	ENVIRONMENT=Server	This setting defines a macro that is environment specific.
R2C_OUT_EXT	.cob	This setting specifies the extension to use for all generated programs.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\cobol.tab	This setting specifies the location of the code generator data file.

CodeGen

The settings in [CodeGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility and are language and platform independent.

Table 2-10 Hps.ini [CodeGen] section

Key	Possible Values / Default	Description
R2C_BINDFILE_DIR	{install_dir}\bnd	This setting specifies the directory to find the rule bind file.

R2C_BLD_DIR	{install_dir}\ad\cfg\cg	This setting specifies the directory to find temporary BLD files.
R2C_CODEGEN_DIR	{install_dir}\ad\cfg\cg	This setting specifies the location of other code generation data files. See also R2C_TABS_DIR.
R2C_ERR_DIR	{install_dir}\temp	This setting specifies the directory to create the temporary error list files.
R2C_INCLUDE_DIR	{install_dir}\nt\sys\inc (default)	This setting specifies the directory for include files. If the directory is specified in the code generation parameter, this setting is overwritten. Refer to the Code Generation Parameters and Settings section of the *Rules Language Reference Guide* .
R2C_LST_DIR	{install_dir}\temp	This setting specifies the directory to create a listing with the rule preparation results.
R2C_MSG_FILE	{install_dir}\ad\cfg\cg\cg.MSG	This setting specifies the location of the codegen error message file.
R2C_OUTPUT_DIR	{install_dir}\temp	This setting specifies the directory to create output files.
R2C_PNL_DIR	{install_dir}\temp	This setting specifies the directory to find window panel files.
R2C_RC_DIR	{install_dir}\temp	This setting specifies the directory to create the RC files.
R2C_SOURCE_DIR	{install_dir}\temp	This setting specifies the directory to find the rule source.
R2C_TABS_DIR	{install_dir}\temp	This setting specifies the location of the code generation data files. See also R2C_CODEGEN_DIR.
R2C_VW_DIR	{install_dir}\temp	This setting specifies the directory to create the VW files.
R2C_WORK_DIR	{install_dir}\temp	This setting specifies the directory to create all temporary work files.
SIZEOF		For an occurring view the SIZEOF function returns the size of one occurrence in ClassicCOBOL, OpenCOBOL and Java modes. In C mode, the size of the entire occurring view is returned. Without this flag, the view size is NOT multiplied by number of occurrences. When this flag is specified, the result is always multiplied by number of occurrences.

CodegenPragmas

The settings in the [CodegenPragmas] section are used by the Code Generation facility and are language and platform independent. This section can be manually added by the user, and contains the Pragmatic statements. For example:

```
[CodegenPragmas]
<pragma_line>
...
<pragma_line>
```

where pragma_line has correct Rules Language syntax. See *Rules Language Reference Guide* for the Pragmatic statement syntax.

CodegenParameters

The settings in the [CodeGenParameters] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and control how the code is generated. The settings in this section are language and platform independent.

Table 2-11 Hps.ini [CodegenParameters] section

Key	Possible Values / Default	Description

ALWAYS_GENERATE_USER_TRACE	YES NO	This setting and command line flag affect Java generation only. When set to YES or the command-line flag GENTRACE is specified for codegen, all TRACE statements from the rule source code are generated. When set to NO and codegen command-line flag GENTRACE is not specified, no user TRACE statements are generated in the Java code. When debugging info generation is ON (-yd command-line parameter is not specified for codegen), all user TRACE statements from the rule source code are generated.
ASSERT_VIEW_IDENTITY	YES (default) NO	This setting specifies if CodeGen generates assertIdentity calls to ensure the use of the proper version of the views in the rule. The parameter also specifies whether to generate getHash() method in view classes. The default value is YES. The setting is only applicable for Java generation.
AUTOCOMMIT	true false	This setting is supported for Java only. By default, DB2 creates a connection with AutoCommit set to TRUE that disables all transaction control. To enable transaction control with SQL statements COMMIT and ROLLBACK connection method, setAutoCommit(false) must be invoked. Generated Java code will have setAutoCommit() call with a parameter equal to the value specified by AUTOCOMMIT (case-sensitive). If AUTOCOMMIT key is not set, then no call will be generated.
BASE_CONTROL_CLASS	Bphx.Sdf.Client.Gui.BphxWindow	This setting specifies the base class of the user control class generated for every AppBuilder window. It is used for C# generation only.
CHECK_DEC_FORMAT	YES (default) NO	This setting controls numeric format string verification. Yes enables verification. No disables verification.
CompareDatesAsString	Y N	This setting is supported for OpenCOBOL only. This setting controls how date comparisons are generated. • Y: Date fields are compared as strings where the result is defined by the environment. • N: Date fields are converted to integer values and compared. This comparison is platform independent. In AB203x, the string comparison is used only in host variables. In all other cases, the date values are converted to integer values and then compared.

COPYFROM_NULL_PARAMETERS_THRESHOLD	10 (default)	This setting allows configuring Java generation of map view to view statement. The parameter value is a threshold percentage for determining a way of views mapping generation. The first option is to use view copyFrom method with a list of view fields. It is used when the percentage of destination view fields which have no corresponding field in the source view (i. e. passed as null to copyFrom method) is less then this parameter value. Otherwise the second option is chosen: field-to-field assignment is generated directly in the rule. Set this parameter to 100 for best generated code maintainability. Set it to 0 for best performance.
DATA_CONVERTER	appbuilder.util.AbfDefaultDataConverter (default) <qualified java class name>	This specifies the Java class implementing the appbuilder.util.AbfDataConverter interface. This class is used for converting data (views, fields) to a char[] array and reverse. Data conversion is performed for OVERLAY and REDEFINE operations. Also data converter counts size of data as a size of its converted representation. The result of size count is the result of Rules SIZEOF std function.
DB2DTFMT	%Y-%0m-%0d	This is valid for C and ClassicCOBOL only. This format is used to convert date fields when calling DB2.
DB2TMFMT	%0t:%0m:%0s	This is valid for C and ClassicCOBOL only. This format is used to convert time fields when calling DB2.
DEFAULT_CENTURY	1900 (default)	This setting is supported for OpenCOBOL only. This is the value for the century in the DATE function when only two digits are used for the year in the input and format strings, and this value is added to the two digit year when parsed. This value is also passed to the support library functions. To change this value for runtime, the rule must be re-prepared. This setting can be overridden by PRAGMA CENTURY statement. See also DEFAULT_CENTURY setting in the [NC] section of the Appbuilder.ini for Java, and DEFAULT_CENTURY in the [AE Runtime] section.
DFLTDTFMT	%0m/%0d/%Y	This is valid for C, ClassicCOBOL and OpenCOBOL. This format is used for DATE and CHAR functions when the second parameter was omitted. For Java, DEFAULT_DATE_FORMAT setting in the [NC] section of the Appbuilder.ini file is used.
DFLTTMFMT	%0t:%0m:%0s	This is valid for C, ClassicCOBOL and OpenCOBOL. This format is used for TIME and CHAR functions when second parameter was omitted. For Java, DEFAULT_TIME_FORMAT setting in the [NC] section of the Appbuilder.ini file is used.
DFLTTSFMT	%0o-%0d-%Y.%0t.%0m.%0s.%f	This is valid for OpenCobol only. This format is used for timestamp and char functions when the second parameter is omitted.

EXPAND_RULE_SIGNATURE	YES NO (default)	This is valid for Java only. Rule and subrule signatures are expanded when possible. Possible Values are YES and NO (default value).
FLOATING_POINT_STANDARD	IEEE754 (default value) HEXADECIMAL	This setting is valid for generations supporting floating-point types (Java). It specifies floating-point type ranges. Possible values are IEEE754 (default value) or HEXADECIMAL.
GENERATE_IO_VIEW_TRACE	YES NO (default)	This setting affects Java generation only. It is set to generate input/output view debug TRACE statements. This setting can have the values YES or NO. When set to YES the input view trace is generated at the beginning of the rule and the output view trace is generated at the end of the rule. The input view is written to the trace on entry to the rule, and the output view, on exit of the rule.
GENERATE_RULE_CALLS_TRACE	YES NO (default)	This setting affects Java generation only. When GENERATE_RULE_CALLS_TRACE is set to YES debug level, TRACE statements are generated at the beginning and end of each rule. When it is set to NO, debug level TRACE statements are not generated. Debug level TRACE statements are generated conditionally, so these statements can be disabled at runtime, if the debug option is turned off.
GENERATE_STATELESS_RULE	YES NO (default)	This is valid for Java only. Rule is generated stateless when possible. Possible Values are YES and NO (default value).
GENERATE_VIEW_FIELD_ACCESSORS	YES NO (default)	This setting is valid for Java only. Public field accessors will be generated for views. Possible Values are YES and NO (default value).
INDEX_CONTROL_ON	YES (default) NO	This setting is supported for C only. This setting specifies whether or not the generated C code performs index checking when accessing the occurring views. See also INDEX_CONTROL_ABORT in the [AE Runtime] section.
INLINE_VIEW_COPY	YES NO (default)	If is set to YES, then the sequence of Java statements corresponding to rules MAP statement with view arguments is generated in the rule class as is (inline). If is set to NO, then this sequence is enclosed to a private rule class method and MAP is generated as call to the method (see also INLINE_VIEW_COPY_FIELDS_LIMIT).
INLINE_VIEW_COPY_FIELDS_LIMIT	2, (default) <number>	When INLINE_VIEW_COPY is NO this number determines the maximum number of fields in a view V for which statement MAP ... TO V is generated inline. If number of fields exceeds this limit, then the Java code corresponding to MAP with destination view V is enclosed to a private rule class method (see also INLINE_VIEW_COPY).

JAVA_PERSISTENT_CURSOR	YES NO (default)	This setting controls the SQL cursor persistence. If it is set to YES, then an SQL cursor will be persistent. Persistence means that a cursor created by a particular rule must be retained after the end of the rule invocation and made available to subsequent invocations of that rule within the same scope until explicitly closed, or in the case of executing in a server request scope, the request terminates. If it is set to YES it is equivalent to setting command line flag SQLPERSIST. Possible values are YES and NO (default value).
LAZY_INSTANTIATION_ENABLED	YES NO (default)	This parameter affects Java generation only. If this parameter is set to YES, all local variables and views (excluding redefined views and input/output views) are instantiated when they are explicitly accessed for the first time. Refer to the Code Generation and Parameters Settings section of the *Rules Language Reference Guide* .
MAX_SUMMARY_IO_VIEW_FIELDS	255	This setting is valid for Java only. It sets maximum number of input and output views fields (summary) for which expanding is provided.
PARAM	<list_of_parameters>	This is valid for C, ClassicCOBOL and OpenCOBOL only. This setting is used to list additional code generation parameters. This list is added to the end of the command line parameters. For example, 'PARAM=-yk' provides backward compatibility. Refer to the Code Generation and Parameters Settings section of the *Rules Language Reference Guide* .
READ_ONLY_SQL_CURSOR	YES NO (default)	This parameter configures a way of generation of an SQL cursor which has no explicit FOR UPDATE specification. When set to YES, an SQL cursor having no explicit FOR UPDATE specification is generated as read-only sqlj iterator. If there is an attempt to perform an UPDATE or DELETE operation positioned by this cursor in the rule code, the code generation submits an error. An SQL cursor which has an explicit FOR UPDATE clause in its declaration is generated as an updatable sqlj iterator. When set to NO, an SQL cursor having no explicit FOR READ ONLY (FETCH ONLY) specification is generated as an updatable sqlj iterator. Cursor explicitly declared as READ ONLY is generated as a read-only iterator.
SQLINITFLAG	0	This setting is supported for OpenCOBOL only. This controls whether SQL-INIT-FLAG will be reset to 0 in each rule. If setting is equal to 0 then MOVE ZERO TO SQL-INITFLAG is generated for each database rule. By default SQL-INIT-FLAG is not initialized. When COBOL SQL co-processor for DB2 is used this flag should be removed (disabled).
TIMEFMT	%0t.%0m.%0s.%0f	This setting is supported for OpenCOBOL only. This format is used when a support library function is called. Only the delimiter can be changed by the end user

TIMEINIT	00.00.00.000	This setting is supported for OpenCOBOL only. This value is used to initialize TIME type fields.
TIMESTAMPDB2CMP	0001-01-01-00.00.00.000000	This setting is supported for OpenCOBOL only. This is the value to compare all host TIMESTAMP variables with. See also TIMESTAMPDB2DFLT.
TIMESTAMPDB2DFLT	0001-01-01-00.00.00.000000	This setting is supported for OpenCOBOL only. This is the value to assign to a TIMESTAMP host variable if it is less than TIMESTAMPDB2CMP.
TIMESTAMPINIT	0000-00-00-00.00.00.000000	This setting is supported for OpenCOBOL only. This value is used to initialize TIMESTAMP type fields.

AB3.0 introduced the following settings that need more than one environment: GLOBAL_PACKAGE, VIEW_PACKAGE, SET_PACKAGE, COMPONENT_PACKAGE.
They were moved to the [AP JAVA] and [AP DOTNET] sections.
AB3.0 introduced SYSTEM_VIEW_PACKAGE, needing more than one environment. It was split into the following: SYSTEM_VIEW_PACKAGE and SYSTEM_SET_PACKAGE. They were also moved to the [AP JAVA] and [AP DOTNET] sections.

For rules created with AB30, the SYSTEM_VIEW_PACKAGE, GLOBAL_PACKAGE, SET_PACKAGE, VIEW_PACKAGE, and COMPONENT_PACKAGE settings have no effect because the package name is now set in the bindfile for every repository object. However, these settings are used by Codegen when it translates rules for which bindfiles were created by earlier versions of AppBuilder.

These changes also apply to AB3.1.

CONVERTER.CASSICCOBOL

The settings in [CONVERTER.CLASSICCOBOL] section handle performance marshalled ClassicCOBOL.

Table 2-12 Hps.ini [CONVERTER.CLASSICCOBOL] section

Key	Possible Values / Default	Description
MODULE	$PATH/bpcc.dll	
TARGET_CP	ibm-037	This setting specifies the codepage the data should be converted into. If it is the same as the local system codepage, the codepage conversion step is skipped.

CONVERTER.OPENCOBOL

The settings in [CONVERTER.OPENCOBOL] section handle performance marshalled OpenCOBOL.

Table 2-13 Hps.ini [CONVERTER.OPENCOBOL] section

Key	Possible Values / Default	Description
DATE_SEPERATOR		This setting is commented out by default. If specified, it overrides the default '-' separator in dates and the date component of timestamp.
MODULE	$PATH/bpcc.dll	
TARGET_CP	ibm-037	This setting is commented out by default. It specifies the codepage the data should be converted into. If it is the same as the local system codepage, the codepage conversion step is skipped.

TIME_SEPERATOR		This setting is commented out by default. It performs similarly with DATE_SEPERATOR, but for time fields.

CClientGen

The settings in [CClientGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the C server generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-14 Hps.ini [CServerGen] section

Key	Possible Values / Default	Description
MACRO	LANGUAGE=C	This setting defines a macro that is language specific.
MACRO	ENVIRONMENT=GUI	This setting defines a macro that is environment specific.
MACRO	CLIENT_MODEL=CONVERSE	This defines a macro that is used to differ a client model
MACRO	PLATFORM=Windows	This defines a macro that is platform specific.
R2C_OUT_EXT	.pcc	This setting specifies the extension to use for all generated programs.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\c.tab	This setting specifies the location of the code generator data file.

CServerGen

The settings in [CServerGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the C server generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-14 Hps.ini [CServerGen] section

Key	Possible Values / Default	Description
MACRO	LANGUAGE=C	This setting defines a macro that is language specific.
MACRO	ENVIRONMENT=Server	This setting defines a macro that is environment specific.
MACRO	PLATFORM=Windows	This defines a macro that is platform specific.
R2C_OUT_EXT	.pcc	This setting specifies the extension to use for all generated programs.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\c.tab	This setting specifies the location of the code generator data file.

CSharpBatchGen

The settings in the [CSharpBatchGen] section contain values that will overwrite any settings from the common sections and are used only for C#.

Hps.ini [CSharpBatchGen] section

Key	Possible Values / Default	Description
R2C_STANDARD_TABS	C:\AppBuilder\ad\cfg\cg\csharp.tab	Specifies the location of the code generator data file.
MACRO	LANGUAGE=CSharp	This defines a macro that is language specific
MACRO	ENVIRONMENT=Batch	This defines a macro that is environment specific
MACRO	PLATFORM=DotNetBatch	This defines a macro that is platform specific.

CSharpClientGen

The settings in the [CSharpClientGen] section contain values that will overwrite any settings from the common sections and are used only for C#.

Table 2-15 Hps.ini [CSharpClientGen] section

Key	Possible Values / Default	Description
R2C_OUT_EXT		This setting specifies the extension to use for all the generated programs.
MACRO	LANGUAGE=CSharp	This setting defines a macro that is language specific.
MACRO	ENVIRONMENT=GUI	This setting defines a macro that is environment specific.
MACRO	SERVLET=FALSE	This setting defines a macro that is servlet specific.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\csharp.tab	This setting specifies the location of the code generator data file.
SC_MT_IMPL	AUTOCLSR (default) SYNC UNSYNC	This setting specifies codegen to generate thread-safe code (SYNC) or non-thread-safe code (UNSYNC), or decide automatically - sync for client and unsync for server (AUTOCLSR).

CSharpCommonGen

The settings in the [CSharpCommonGen] section contain values that will overwrite any settings from the common sections and are used only for C#.

Hps.ini [CSharpCommonGen] section

Key	Possible Values / Default	Description
BASE_CONTROL_CLASS	Bphx.Sdf.Client.Gui.BphxWindow	Specifies the base class of the user control class generated for every AppBuilder window. Used for CSharp generation only. This setting is required and must specify a valid class that will be accessible during rule preparation.
PARAM	-fFORCEMAIN	Specifies generation of the Main() method in the module class.
BASE_CONTROL_CLASS_WPF	Bphx.Sdf.Client.Gui.SdfBphxWindow	Specifies the base class of the user control class generated for every AppBuilder window. Used for CSharp WPF generation only.

CSharpServerGen

The settings in the [CSharpServerGen] section contain values that will overwrite any settings from the common sections and are used only for C#.

Hps.ini [CSharpServerGen] section

Key	Possible Values / Default	Description
R2C_STANDARD_TABS	C:\AppBuilder\ad\cfg\cg\csharp.tab	Specifies the location of the code generator data file.
MACRO	LANGUAGE=CSharp	This defines a macro that is language specific
MACRO	ENVIRONMENT=Server	This defines a macro that is environment specific
MACRO	PLATFORM=DotNetServer	This defines a macro that is platform specific.

DBE

The settings in [DBE] section are used for the Construction Workbench and Workgroup Repository.

Table 2-16 Hps.ini [DBE] section

Key	Possible Values / Default	Description
DBE	0	This setting is for the backward compatibility, and used in HPS Personal Repository.

FREEWAY

The settings in [FREEWAY] section are used for the Construction Workbench and Workgroup Repository. For information about the data migration, refer to the *Repository Administration Guide* .

Table 2-17 Hps.ini [FREEWAY] section

Key	Possible Values / Default	Description
AUTO_TOGGLE_FILES	TRUE (default) FALSE	When set to TRUE, the create setting for an entity's files and relationship match that of the entity. When set to FALSE, you must manually toggle off the create setting for each entity file and relationship when an entity's create setting is toggled off.
AUTO_TOGGLE_RELATIONS	TRUE (default) FALSE	When set to TRUE, the create setting for an entity's relationship matches that of the entity. When it is set to FALSE, you must manually toggle off the create setting for each entity relationship when an entity's create setting is toggled off.
BEEP	TRUE FALSE (default)	This setting specifies whether or not to sound the beep when the data migration export is completed.
CLIENT_DNARPC_TIMEOUT	300	This setting specifies the amount of time in seconds a DNA client connection will wait for a response before a time-out occurs.
COMPARE_OFF	TRUE (default) FALSE	This setting specifies whether or not to compare objects being imported with the objects in the target repository during the data migration import.
CONFIRM_COMMIT_ENABLED	TRUE (default) FALSE	When set to TRUE, Freeway Explorer prompts the user if they are sure they want to commit the changes. If set to FALSE, changes are committed without the prompt when the commit button is pressed.
DBCS_MIGRATIONS	TRUE FALSE	This setting is required if objects or source is exported from a personal (local) repository or Workgroup (Freeway) repository that contains DBCS characters. This setting is set on install. When set to TRUE, the migration imports adjust the byte count for download from the host. It is set to TRUE when installing in a DBCS environment.
EXPORT_DISPLAY_LOG	TRUE FALSE (default)	This setting specifies whether or not to automatically display the LOG file after export.
FILEDIFF	COMPFILE.EXE (default)	This setting specifies the executable file that is used to compare files in the data migration.
FILE_DIR	{install_dir}\FILES	This setting specifies the root directory where Workgroup/Personal (local) Repository files are stored.
FWY_FILE_DEF	{install_dir}\AD\CFG\DATA\FWYFILE.DEF	This setting specifies the definition file for fwyutil (import/export) describing which file types are considered binary and which are considered ASCII.
IMPORT_DEFAULT_OBJ	TRUE FALSE (default)	When set to FALSE, AppBuilder will not attempt to import default objects if they exist in the repository already. Setting to TRUE will treat default objects as any other and attempt to analyze and import.
LVLEXP_AS_HIERARCHY	TRUE (default) FALSE	This setting dictates how a LEVEL MIGRATION EXPORT functionality from the Personal or Workgroup Repository acts on import. • TRUE: The migration exports the entire hierarchy. • FALSE: The migration exports all objects and relationships in the level selected as Entity Only.
MIG_ABORT_ON_ERROR	TRUE (default) FALSE	When set to TRUE, Explorer will stop processing the migration import if an error occurs. If set to FALSE, Explorer will attempt to continue even though an error occurred.
MIG_GLOBAL	,=,,,<	This setting specifies the filter data used by Explorer for exporting migrations.
MIGRATION_DIR	{install_dir}\MIGFILES	This setting specifies the directory where Freeway Explorer migration files are kept. Explorer will write the directory it was last using out to this key so it can be opened in the same directory next time.

MIGRATION_TRACE	TRUE FALSE (default)	This setting specifies whether or not to generate a string for every routine the migration calls and places it in the log file after export.
TEMPORARY	{install_dir}\TEMP	This setting specifies the temporary storage location used by some AppBuilder calls. For example, this directory becomes the working directory to store files during the creation of a DB2 database and import of the data.
USE_VERSION_GROUP	YES	This setting is for the backward compatibility, and used in HPS versioning.
VER_COPY_DIR	{install_dir}\FWYMGNT\VERCOPY	This setting specifies the location for temporary files needed during a version copy.

_FREEWAY_DLLS_

The settings in [FREEWAY_DLLS] section are used for the Construction Workbench and Workgroup Repository.

Table 2-18 Hps.ini [FREEWAY_DLLS] section

Key	Possible Values / Default	Description
MODEL	dynnt	This setting specifies the model specific file.

_FREEWAY_SERVER_

The settings in [FREEWAY_SERVER] section are used for the Construction Workbench and Workgroup Repository.

Table 2-19 Hps.ini [FREEWAY_SERVER] section

Key	Possible Values / Default	Description
DBM_DATABASE	FREEWAY	This setting specifies the latest repository that has been accessed.
INCREMENTAL_BUILD	TRUE (default) / FALSE	When set to TRUE, the repository tracks objects and determines which are "Prepared" meaning already built, or "Unprepared" meaning they need to be built.
OBJECT_QUERY_LIMIT	<integer>	This setting applies to the Personal Repository and the Freeway Workgroup Server (not client). This setting specifies the query limit for the number of objects returned to the client. For example, if there are 180,000 rules in the repository and you set 5,000 for this setting, the query returns only the first 5,000 rules. You must manually add this setting.
PRE_SPAWN_SERVERS	0	This setting specifies the number of servers that are listening for users to connect to the repository.
REBUILD_ENABLED	TRUE FALSE (default)	When set to TRUE, the Rebuild feature of the Workgroup Repository is enabled. Default value is FALSE.
SQLQRY_CACHE_ENABLED	TRUE (default) FALSE	When set to TRUE, it enables the Query Caching that can help migration performance when working against Windows Repository.
TIP_DIR	{install_dir}\TIPFILE\TIPFILE.XXX	This setting specifies the file that tracks session activity.

JavaClientGen

The settings in [JavaClientGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the Java generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-20 Hps.ini [JavaGen] section

Key	Possible Values / Default	Description
MACRO	LANGUAGE=Java	This setting defines a macro that is language specific.

MACRO	ENVIRONMENT=GUI	This setting defines a macro that is environment specific.
R2C_OUT_EXT	.java	This setting specifies the extension to use for all generated programs.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\java.tab	This setting specifies the location of the code generator data file.

JavaBatchGen

This section contain values that will overwrite any settings from the common sections and is used only for Java.

Table 2-21 Hps.ini [JavaBatchGen] section

Key	Possible Values / Default	Description
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\javab.tab	Specifies the location of the code generator data file.
R2C_OUT_EXT	.java	Specifies the extension to use for all generated programs.
MACRO	=LANGUAGE=Java	Defines a macro that is language specific.
MACRO	=ENVIRONMENT=Batch	Defines a macro that is environment specific.

JavaHPServerGen

The settings in [JavaHPServerGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the Java server generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-23 Hps.ini [JavaServerGen] section

Key	Possible Values / Default	Description
AUTOCOMMIT	TRUE FALSE (default)	By default DB2 creates a connection with AutoCommit set to true which disables all transaction control. To enable transaction control with SQL statements COMMIT and ROLLBACK connection method setAutoCommit(false) must be invoked. Generated Java code will have setAutoCommit() call with a parameter equal to the value specified by AUTOCOMMIT (case-sensitive). If AUTOCOMMIT key is not set then no call will be generated
DFLTDTFMT	%0m/%0d/%Y	This format string is used for DATE and CHAR functions when second parameter was omitted.
DFLTTMFMT	%0t:%0m:%0s	This format string is used for TIME and CHAR functions when second parameter was omitted.
DFLTTSFMT	%0o-%0d-%Y.%0t.%0m.%0s.%f	This format string is used for TIMESTAMP and CHAR functions when second parameter was omitted.
JAVA_PERSISTENT_CURSOR	YES NO(default)	Controls SQL cursor persistance. YES or NO default is NO.
MACRO	LANGUAGE=Java	This setting defines a macro that is language specific.
MACRO	ENVIRONMENT=Server	This setting defines a macro that is environment specific.
MACRO	PLATFORM=JavaHPServer	This defines a macro that is platform specific.
PARM	PARAM=<list_of_parameters>	This overwrites the PARAM setting in the [CodegenParameters] section.
PARM	<ParameterName>=<parameter_value>	This overwrites the setting in the [CodegenParameters] section with the name *ParameterName,* giving it the value *parameter_value* .
R2C_OUT_EXT	.java	This setting specifies the extension to use for all generated programs.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\java.tab	This setting specifies the location of the code generator data file.

READ_ONLY_SQL_CURSOR	YES NO(default)	This parameter controls the reaction of Codegen on attempt to modify data under read only SQL cursors (with no FOR UPDATE clause in declaration). Possible values: YES - Codegen generates an error if there is an attempt to perform an UPDATE or DELETE operation positioned by a READ ONLY cursor in the code. NO - Codegen generates an additional FOR UPDATE clause in read only cursor declarations. (default)
SET_TEMPLATE_DIR	tmpl	Specifies the package folder for set templates, the folder is created under R2C_OUTPUT_DIR. Default folder name is "tmpl".
VIEW_TEMPLATE_DIR	tmpl	Specifies the package folder for view templates, the folder is created under R2C_OUTPUT_DIR. Default folder name is "tmpl".

JavaHTMLGen

The settings in [JavaHTMLGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the Java HTML generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-22 Hps.ini [JavaHTMLGen] section

Key	Possible Values / Default	Description
MACRO	LANGUAGE=Java	This setting defines a macro that is language specific.
MACRO	ENVIRONMENT=HTML	This setting defines a macro that is environment specific.
PARM	PARAM=<list_of_parameters>	This overwrites the PARAM setting in the [CodegenParameters] section.
PARM	<ParameterName>=<parameter_value>	This overwrites the setting in the [CodegenParameters] section with the name *ParameterName* , giving it the value *parameter_value* .
R2C_OUT_EXT	.java	This setting specifies the extension to use for all generated programs.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\javah.tab	This setting specifies the location of the code generator data file.

JavaServerGen

The settings in [JavaServerGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the Java server generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-23 Hps.ini [JavaServerGen] section

Key	Possible Values / Default	Description
MACRO	LANGUAGE=Java	This setting defines a macro that is language specific.
MACRO	ENVIRONMENT=Server	This setting defines a macro that is environment specific.
PARM	PARAM=<list_of_parameters>	This overwrites the PARAM setting in the [CodegenParameters] section.
PARM	<ParameterName>=<parameter_value>	This overwrites the setting in the [CodegenParameters] section with the name *ParameterName,* giving it the value *parameter_value* .
R2C_OUT_EXT	.java	This setting specifies the extension to use for all generated programs.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\java.tab	This setting specifies the location of the code generator data file.

JobScheduler

The settings in [JobScheduler] section are used for the Construction Workbench.

Table 2-24 Hps.ini [JobScheduler] section

Key	Possible Values / Default	Description
DefaultPrepProtocol	tcpip	This setting specifies the protocol used for Remote Preparation.
JobFileName	{install_dir}\JOBFILE	This setting specifies the location of the Prepjob files.
MF_JOB_SUBMIT_WAIT_INTERVAL	30 (default)	When a mainframe job has been submitted by a "Client Side Codegen" preparation, Preparation will wait for the results of the mainframe job to be posted to the "Prep Status Log". This setting specifies the number of seconds that Preparation is to wait before it begins to poll the mainframe to see if the job has completed without posting its results to the "Prep Status Log". The main reason a job fails to post results is a JCL error. This error can be caused by a configuration error such as an invalid data set name for a CICS or DB2 subsystem.
MF_JOB_POLLING_INTERVAL	5 (default)	This setting specifies the number of seconds that Preparation waits between polling events. Once a mainframe job has been submitted by a "Client Side Codegen" preparation and the number of seconds defined by MF_JOB_SUBMIT_WAIT_INTERVAL has elapsed for that job, Preparation will then begin polling JES to find jobs which have completed in error.
MF_JOB_FILE_LIMIT	2KK (default) -1	This setting specifies the maximum number of bytes to be recorded by Preparation from the output of a mainframe job that failed to post its results. When the JES output exceeds the value specified for this setting, the output will be truncated to this size and then recorded by Preparation in the Job's "Prep Status Log". When -1 is specified, there is no limit to the amount of JES output recorded by Preparation in the Job's "Prep Status Log".
MF_JOB_POST_SUBMIT_EXIT	(no default)	This setting specifies the user-defined executable that will be invoked for every job submitted to the mainframe by Preparation. The executable can be a fully qualified module name. If it is not fully qualified, it must be qualified with its correct extension such as .exe and .cmd, and reside in a directory defined in the PATH environment variable. The executable can be defined with any number of arguments; however, it must be able to accept the following additional arguments: • ? *host hostname/ipaddr* : This specifies the mainframe to which the job was submitted. • *? job nnnnn* : This specifies the five digit JES job name that was submitted. For example, *MF_JOB_POST_SUBMIT_EXIT=myPostSubmitExit.cmd* " *myargument* " The resultant "command line", for the above example, invoked by the Preparation when job number "01234" is submitted to the host mainframe "fred" will be: *myPostSubmitExit.cmd* "myargument" ? *host fred* ? *job 01234* Note: The executable must not be a GUI module. The GUI module will not execute.
OutputDir	{install_dir}\DBG	This setting specifies the location of Preparation status files.
Trace	Info Warnings Errors Info+Warnings Info+Errors Errors+Warnings Errors+Warnings+Info	This setting specifies the preparation status trace level. You can specify any combination of error, warning, or informational messages.

MacroDefinition

The [MacroDefinitions] section defines macros that can be used for all target languages and platforms. These settings are used by the Code

Generation facility and are language and platform independent. The [MacroDefinitions] section can be viewed and updated from **Construction Workbench > Tools > Workbench Options**, **Preparation** set of options, **Conditionals** button.

MacroDomains

The settings in the [MacroDomains] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility and are language and platform independent.

Table 2-25 Hps.ini [MacroDomains] section

Key	Possible Values / Default	Description
ENVIRONMENT	Server,HTML,GUI,Batch	This setting specifies all the possible values for macro ENVIRONMENT.
LANGUAGE	Java,C,Cobol,OpenCobol	This setting specifies all the possible values for macro LANGUAGE.

NLS

The settings in [NLS] section are used for the Construction Workbench and Workgroup Repository, Java Client, HTML Client, EJB, SOAP Web Services, Java Reports.

Table 2-26 Hps.ini [NLS] section

Key	Possible Values / Default	Description
Codepage_AIX		This setting specifies the codepage setting for UNIX machine. This value is used for remote preparation to UNIX.
Codepage_WIN	1004 (default) 1253 932 949 874	This setting specifies the codepage setting for Windows machine. It identifies the codepage to use at runtime and preparation. Codepage_WIN setting is the same as the number given in the [AE Runtime] section LANGUAGE setting. • 1004: US-English, Danish, German, English, Spanish, French, Italian, Portuguese • 1253: Greek • 932: Japanese • 949: Korean • 874: Thai When installing in a Japanese environment, it is set to 932. When installing in a Korean environment, it is set to 949.

OpenCobolGen

The settings in [OpenCobolGen] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and apply to the OpenCOBOL generation only. These settings will overwrite settings from the language and platform independent sections where applicable.

Table 2-29 Hps.ini [OpenCobolGen] section

Key	Possible Values / Default	Description
MACRO	LANGUAGE=OpenCobol	This setting defines a macro that is language specific.
MACRO	ENVIRONMENT=Server	This setting defines a macro that is environment specific.
R2C_OUT_EXT	.cbl	This setting specifies the extension to use for all generated programs.
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\ocob.tab	This setting specifies the location of the code generator data file.

_PERSONAL_REPOSITORY_

The settings in [PERSONAL_REPOSITORY] section are used for the Personal Repository.

Table 2-30 Hps.ini [PERSONAL_REPOSITORY] section

Key	Possible Values / Default	Description
CREATE_LRE_ON	C:	This setting specifies the location where the database for the personal repository is created.
FTP_CUSTOM_QUOTE		Add this setting to issue a custom command for an FTP session during the upload/download processing when using the Personal Repository. This setting might be necessary when converting currency symbols or codepage alias. By setting this option to a specific argument, you can issue the command during the FTP session on the host server. For example, • FTP_CUSTOM_QUOTE=site sbd=tcpip.dutch.tcpxlbi • FTP_CUSTOM_QUOTE=site sbdataconn=(ibm-285, cp1252).
FTP_TRACE	TRUE FALSE (default)	This setting specifies whether or not to create a file for tracing of repository operations during uploads and downloads. When set to TRUE, it creates a trace file LRE_FTP.trace in the nt\sys\bin directory under where AppBuilder is installed. LRE_FTP.trace file contains the application logic flow of FTP commands used during upload and download. WINSOCK_TRACE must also be set to TRUE.
LRE_DOWNLOAD_DIR	{install_dir}\DOWNLOAD	This setting specifies the location where all the files necessary for download are stored.
LRE_DOWNLOAD_TYPE	All AsmanyAsPossible (default) Nothing	This setting specifies how the download procedure performs the local import.
LRE_OPERATIONS_AWARE	TRUE (default) FALSE	When set to TRUE, it does not allow the Construction Workbench to open if an upload or download is in progress. When set to FALSE, the Construction Workbench can be opened even during the upload or download process.
LRE_UPLOAD_DIR	{install_dir}\UPLOAD	This setting specifies the location where all the files necessary for upload are stored.
LRE_UPLOAD_TYPE	All AsManyAsPossible (default) Nothing	This setting specifies how the upload procedure performs the remote import.
SEND_SBDATACONN_VALUE	TRUE (default) FALSE	When set to TRUE, it presets the FTP session with the mainframe to the user's specific codepage settings and ignores what the mainframe default is. It sends the codepage value to the FTP server for each FTP session, asking the server to not act upon a static setting, but rather use the value supplied with SBDATACONN.
USER_REMOVEABLE_UOW	TRUE (default) FALSE	This setting enables the menu item that removes the selected items in the Upload window.
WINSOCK_TRACE	TRUE FALSE (default)	This setting specifies whether or not to create a file for tracing the communication part of the repository operations during uploads and downloads. When set to TRUE, it creates a trace file in the nt\sys\bin directory under the AppBuilder install directory. FTP_TRACE must also be set to TRUE. Default value is FALSE.

REPOSITORIES

This section is created after the repository creation, using the Repository Administration tool, and contains information about created repositories. It lists the type of repository (personal/workgroup), an ODBC connection statement, the communication protocol being used, the machine the repository resides on, the current repository codepage, and the current state of the repository (Active, Halted). Under normal operation, you should not modify it.

RepositoryAdmin

The settings in [RepositoryAdmin] section are used for the Construction Workbench and Workgroup Repository.

Table 2-31 Hps.ini [RepositoryAdmin] section

Key	Possible Values / Default	Description
ImportRebuildFromEXP	TRUE FALSE (default)	This setting specifies if the rebuild from EXP files is imported so that it can be shared across repositories. Possible values are: TRUE and FALSE (default value).
ImportUOWFromEXP	TRUE FALSE (default)	This setting specifies if the UOW from EXP file is imported, so that it can be shared across repositories. Possible value are: TRUE and FALSE (default value).
DefaultRepository	{install_dir}\AD\DBF\DBBACKUP\ABREPDEF.EXP	This setting specifies the last repository accessed when a restore was done.

Repository Client

The settings in [Repository Client] section are used for the Construction Workbench and Workgroup Repository.

Table 2-32 Hps.ini [Repository Client] section

Key	Possible Values / Default	Description
CACHE_SIZE	100	This setting specifies the number of objects to try and keep in client side cache.
DOM_RESOURCE	GRE437RC.DLL	This setting specifies the DLL containing the domain value external names for objects.
DOMAIN_DIR	{install_dir}\AD\CFG\DATA	This setting specifies the location of domain property files.
LAST_USER_NAME	<user name>	This setting specifies the last user name logged in to the Workgroup Repository or Personal Repository.
METAMOD_DIR	{install_dir}\AD\CFG\DATA	This setting specifies the location of model specific files.
MPANEL_DIR	{install_dir}\AD\CFG\METHOD	This setting specifies the location where the description of the object property panels are stored.
PANEL_EDIT_DIR	{install_dir}\AD\CFG\METHOD	This setting specifies the location of specific edit panel files.
PANEL_FONT	<font size and style> 8.Helv 9.MS UI Gothic	This setting specifies the font used to display object property panels.
PANEL_METHOD_DIR	{install_dir}\AD\CFG\METHOD	This setting specifies the location of specific method panel files.
RECYCLE_DATABASE_COUNT	<integer value>	This setting determines the number of objects that are migrated before an automatic commit. If it is to 0, you must perform a manual commit after the migration has completed
REFRESH_DIR	{install_dir}\TEMP	Legacy setting that is no longer used.
RESOURCE	USALRErc.DLL JPNLRErc.DLL	This setting specifies the DLL containing Workgroup/Personal Repository message strings.
SCOPE_DIR	{install_dir}\AD\CFG\DATA	This setting specifies the location of the scope files that filter the objects that can be seen in specific parts of AppBuilder.

REPOSITORY SERVER

The settings in [REPOSITORY SERVER] section are used for the Construction Workbench and Workgroup Repository.

Table 2-33 Hps.ini [REPOSITORY SERVER] section

Key	Possible Values / Default	Description
UNIQUE_CONTROL_FILE	{install_dir}\FREEWAY.ID	This setting specifies the file that contains the unique Workgroup/Personal Repository ID and seed for creating new objects.

Dna.ini Settings

Dna.ini file contains initialization information for communication when an AppBuilder C Client calls an AppBuilder Server. Refer to Appbuildercom.ini Settings for the communication settings for the Java Client.

A Dna.ini file exists for each client and each server; therefore, there are multiple Dna.ini files in different locations. Each file is used for a specific client or server. Depending on which client or server the file is used for, only the relevant sections of the file are used.

Supported platforms and default locations lists the default locations of the dna.ini file for supported platforms:

Supported platforms and default locations

Supported Platform	Default Dna.ini location
MVS	<HIGH_LEVEL_QUALIFIER>.DNA.INI
UNIX	$DNADIR/nes_apps/<server_name>/dna.ini [clients and servers] $DNADIR/sa/dna.ini [agents, MQSeries queue managers]
Windows	\dna\nt\dna.ini [clients, servers, and gateways] \dna\nt\sa\dna.ini [agents, MQSeries queue managers]

Entries in the Dna.ini files are case sensitive. When using the Dna.ini file for client or server purposes, the path must be set in both the client and server environments before executing the application.

For example:

```
SET DNAINI=<path>
```

Sections in the Dna.ini

The Dna.ini file contains the following sections.

[CWS_SERVER]	[DNA_SERVER]	[MVS_LISTENER]
[CWS_CONVERTER]	[IMPLICIT_EVENTS]	[NLS]
[DNA]	[LU2]	[ROUTING]
[DNA_EXITS]	[LU6.2]	[SMA]
[DNA_PRIVATE]	[MESSAGING]	[TRACING]

{install_dir} in the Possible Value/Default column indicates the AppBuilder installation directory.

CWS_CONVERTER

The settings in [CWS_CONVERTER] section are described in the table below. This section applies to HTTP SUPPORT. The settings pertain only to the CICS environment.

Dna.ini [CWS_CONVERTER] section

Key	Possible Values / Default	Description

<CONVERTER-TYPE>	<Converter-Program>	<Converter-Program> is the name of the converter program to be used for the associated Content-Type. The supported options are: **OPENCOBOL** - Converter for OpenCOBOL view data, associated with application/bpoc **OPENCOBOL-COMPRESSED** - Converter for encrypted/compressed OpenCOBOL view data, associated with application/bpoc-compressed **CLASSICCOBOL** - Converter for ClassicCOBOL view data, associated with application/bpcc **CLASSICCOBOL-COMPRESSED** - Converter for ClassicCOBOL view data, associated with application/bpcc-compressed **ERRORMESSAGE** - Converter for (server-side) error messages, associated with application/bperr. **ERRORMESSAGE-COMPRESSED** - Converter for encrypted/compressed (server-side) error messages, associated with application/bperr-compressed. **TEXT** - Converter for text BLOB data, associated with application/bp-text. **TEXT-COMPRESSED** - Converter for encrypted/compressed text BLOB data, associated with application/bp-text-compressed. **TEXTID** - Converter for text BLOB ID, associated with application/bp-textid. **TEXTID-COMPRESSED** - Converter for encrypted/compressed text BLOB ID, associated with application/bp-textid-compressed. **BINARY** - Converter for binary BLOB data, associated with application/bp-binary. **BINARY-COMPRESSED** - Converter for encrypted/compressed binary BLOB data, associated with application/bp-binary. **BINARYID** - Converter for binary BLOB ID, associated with application/bp-binaryid.

CWS_SERVER

The settings in [CWS_SERVER] section are described in the table below. This section applies to HTTP SUPPORT. The settings pertain only to the CICS environment. HTTP settings are explained in the *Communications Guide* .

Dna.ini [CWS_SERVER] section

Key	Possible Values / Default	Description
BLOBEXIT_TEXT	<BLOB-Exit-Program>	<BLOB-Exit-Program> is the name of the BLOB handler program to be used for the associated BLOB type - TEXT or BINARY. The default value is BPSBLOB.
BLOBEXIT_BIN	<BLOB-Exit-Program>	<BLOB-Exit-Program> is the name of the BLOB handler program to be used for the associated BLOB type - TEXT or BINARY. The default value is BPSBLOB.
TSQ_PREFIX	<TSQ-Prefix>	<TSQ-Prefix> is used to prefix the temporary storage queue names used by the Business Logic to pass large view data. Typically there will be a corresponding TSMODEL definition. The default value is ABCWSTSQ.
TSQ_MAX_ITEMLEN	<Max-Item-Length>	<Max-Item-Length> specifies the maximum length of a single TSQ item ("record"). The default is 8,000 bytes.
CWS_VIEW_IN_MEMORY	Y N	This setting is useful in single region environments as specifying YES causes the front-end to pass large (>32K) view data via addresses in the commarea. Any other value will result in data passed via Temporary Storage queues.

DNA

The settings in [DNA] section are described in the table below.

Dna.ini [DNA] section

Key	Possible Values / Default	Description
CLIENT_DNARPC_TIMEOUT	600 (default)	For clients, this setting specifies the number of seconds without communication from a server. The popping of this timer returns control to the client application with an error number 455. If this variable is set to 0, no timer is activated. ⚠ To prevent a client from timing out before a gateway, make sure the client has a greater time-out value than the gateway.
CLIENT_POLLING_INTERVAL	250 (default)	For clients, this setting specifies the number of milliseconds to wait before polling for data. Values less than 100 milliseconds may adversely affect other executing processes. The supported platform is LU6.2 and Named Pipes connections only.
CLT_MAX_SERVERS	default: 100 (for UNIX) 50 (for all other platforms)	For workstation clients, this setting specifies the number of servers the client can communicate with.
DNA_CLOSE_SEMANTICS	COMMIT (default) ABORT	For clients, this setting specifies how pending database transactions are handled when the client context closes. This is relevant only when the LOG_UNIT_OF_WORK setting is set to CONVERSTATIONAL.
DYNAMIC_DATA_OPTIMIZE	Y (default) N	For clients, this setting specifies whether to transmit an optimized control-structure array with dynamic service requests. An optimized control-structure array does not include all of the control information normally sent to the server (such as field names). This setting must be set to Y to use performance marshalling to optimize the sending of data. Refer to the *Communications Guide* for more information about the performance marshalling. See also the STATIC_DATA_OPTIMIZE setting.

FILE_XFER_LOCATION	{install_dir}\files	For clients and servers, this setting specifies the local destination of files transferred to the current platform.
		
		For workstation platforms, this setting specifies the directory path. AppBuilder assigns a file name unique to the transfer. If the variable is empty, AppBuilder places the received file in the current directory.
		
		For MVC CICS, specify the CICS file name associated with a VSAM KSD. AppBuilder assigns a key unique to each transfer. If this variable is empty on MVS, transfers to that platform fail. The KSDS must be defined as follows:
		
		`DEFINE CLUSTER`
		`(`
		`  NAME(MYKSDS)`
		`  CYLINDER(6, 6)`
		`  INDEXED`
		`  KEYS(32, 0)`
		`  VOLUME(MYVOL)`
		`  RECORDSIZE(8192, 8192)`
		`)`

LOG_UNIT_OF_WORK	NONCONVERSATIONAL (In previous releases, this used to be 52LOCAL) (Default) PSEUDOCONVERSTAIONAL (In previous releases, this used to be LOCAL) CONVERSATIONAL (In previous releases, this used to be REMOTE)	For clients, this setting specifies how pending database transactions are handled. (Values for pervious releases are shown in parentheses.) Use the following values for the desired case: **NONCONVERSATIONAL (52LOCAL)** - You want the server to commit pending database transactions after service request completes and then close the server connection. Use this value for communications with the mainframe. **PSEUDOCONVERSTAIONAL (LOCAL)** - You want the server to commit pending database transactions after the service request completes but leave the server connection open. **CONVERSATIONAL (REMOTE)** - You want the server to commit or abort pending database transactions when the client context closes or as specified explicitly in the code. In the former case, the value of the DNA_CLOSE_SEMANTICS setting determines whether AppBuilder commits or aborts pending transactions. The server connection closes when he client context closes. ⚠ You need not set this variable on a gateway. The client setting is propagated to the gateway and overrides a conflicting value on the gateway. Set [DNA_SERVER]LUW_OVERRIDE to Y on the gateway if you want the gateway value to override the client value.
PROTOCOLS	{install_dir}\nt\rt\DNAPROTO.CFG	For clients and servers, this specifies the path and name of the protocol configuration file.
STATIC_DATA_OPTIMIZE	Y (default) N	For clients, this specifies whether or not to transmit an optimized control-structure array with static service requests, events, and messages. An optimized control-structure array does not include all of the control information normally sent to the server (such as field names). See also DYNAMIC_DATA_OPTIMIZE setting.
VERSION	3.1	
WORKSTATION_ID		For clients, this specifies a string specifying the workstation, per a site-specific naming scheme.

DNA_EXITS

The settings in [DNA_EXITS] section are described in the table below.

Dna.ini [DNA_EXITS] section

Key	Possible Values / Default	Description
DATABASE_EXIT		For servers, this specifies the path and name of an AppBuilder supplied DLL that holds database exits. This is not required for Windows NT.
DNA_AUTHENT_EXIT	STATIC_LINK	For clients, this specifies the path and name of the client authentication exit, which contains logic for eliciting a logon ID and password. In the MVS C370 runtime, specify STATIC_LINK.
DNA_AUTHOR_EXIT	STATIC_LINK	For clients, this specifies the path and name of the client authorization exit, which contains logic for authorizing a client's access to services. In the MVS C370 runtime, specify STATIC_LINK.
DNA_ENCRYPT_EXIT		For clients, this specifies the path and name of the client encryption exit, which contains logic for encrypting a user's password for transmission.
DNA_RPC_END_EXIT	STATIC_LINK	For clients and servers, this specifies the path and name of the trace exit, which contains the RPC-end-exit code used for tracing. In the MVS C370 runtime, specify STATIC_LINK.

SRV_AUTHENT_EXIT	STATIC_LINK	For servers, this specifies the path and name of the server authentication exit, which contains logic for decrypting the user's password and for checking the logon ID and password for validity. In the MVS C370 runtime, specify STATIC_LINK. In MVS, the exit is invoked after CICS checks the client password, so encryption on the client side is inappropriate.
SRV_AUTHOR_EXIT	STATIC_LINK	For servers, this specifies the path and name of the server authorization exit, which contains logic for authorizing a client's access to services. In the MVS C370 runtime, specify STATIC_LINK.
SRV_RTX_EXIT		This specifies the name of the Routing and Transaction Switching exit program. The default is to invoke the supplied DNARTX program, which uses HPSTBLE1/2. See also the [DNA_SERVER] section RTX_EXITDATALEN setting. ⚠ CICS only

DNA_PRIVATE

The settings in [DNA_PRIVATE] section are described in the table below.

Dna.ini [DNA_PRIVATE] section

Key	Possible Values / Default	Description
SERVER_VERSION	2	It is for internal use and should not be changed by the users.
VERSION_UPDATE	0000000000000000	It is for internal use and should not be changed by the users.

DNA_SERVER

The settings in [DNA_SERVER] section are described in the table below. These settings are applicable only to the CICS version.

Dna.ini [DNA_SERVER] section

Key	Possible Values / Default	Description
ALLOW_EXPIRED_RULE	no default	For CICS, this setting specifies the name of a rule that the Listener will invoke even if the provided user ID's password has expired. See also USE_SECURITY_RC.
DUMP_CORE	Y N (default)	For UNIX banking and forking servers, this specifies whether or not to dump the core on receiving SIGSEGV, SIGILL, SIGBUS, SIGPIPE, OR SIGSYS. When set to N and the server process is a forking parent, the process closes any open file/socket descriptors and automatically starts a new forking server with an appropriate message. When set to N and the process is a forking child or banking worker, the process closes any open file/socket descriptors and exits with an appropriate message.
IMPNAME_FILE	<path name> HPSTABLE	For servers, when the USE_IMP_NAMES setting is set to Y, the path and name of a text file containing lines with two case-sensitive fields: the first field is the name of the requested service, the second is the path and name of the DLL to be accessed in response to a request for the service. The first field in each entry begins in column 1. The file may include comment lines, which begin with a pound sign (#) in column 1. This file is automatically populated by AppBuilder if USE_IMP_NAMES = Y. For AppBuilder on the mainframe, specify IMPNAME_FILE=HPSTABLE instead of a path name.

LU_NAMES	<profile_name, profile_name, ...>	For forking servers on LU6.2, this specifies the channels the server listens on. The value must be a comma-separated list of server side LU6.2 profile names.
LUW_OVERRIDE	Y N (default)	For gateways, this setting specifies whether or not the value of the variable [DNA]LOG_UNIT_OF_WORK on the gateway overrides its value on the client.
MVS_SERVICE_TYPE	DNA (for standalone AppBuilder Communications) HPS (for AppBuilder)	For MVS servers, this controls the interface between AppBuilder and an invoked service.
NEMONITOR_CFG	2000:2000,3000: 2000:2000: 60000,5:5:15:5:2	For banking servers, this specifies a string of configuration values in the form: BQ:GQ,T1:T2:T3:T4:N1:N2:N3:N4:N5 where: • BQ is the size of the queue for incoming connections. Unsupported in UNIX; specify 0. • GQ is the size of the queue for used handles to be freed by the garbage collector. Unsupported in UNIX; specify 0. • T1 is the garbage collection interval in milliseconds. Unsupported in UNIX; specify 0. • T2 is the spawning interval; the minimum interval between two successive spawns of worker processes. Unsupported in UNIX; specify 0. • T3 is the connection pickup time in milliseconds. If a connection request waits for more than T3 milliseconds without being picked up by a worker process, a new worker process is spawned to handle the request. The default is 2000. Unsupported in UNIX; specify 0. • T4 is the amount of time in milliseconds on PC hosts and in seconds on UNIX hosts that a dynamically spawned worker process can remain idle before it is killed. The default is 60000 for PC and 60 for UNIX. • N1 is he number of worker processes to be spawned by the banker when started. The default is 5. • N2 is the number of permanent dynamically spawned workers. Permanent workers are never killed. Unsupported in UNIX; specify 0. • N3 is the maximum number of worker processes in the system, including pre-spawned workers and dynamically spawned workers. The default is 15. • N4 is the queuing threshold. The banker spawns additional workers if the number of connection requests reaches this value. The default is 5. Unsupported in UNIX; specify 0. • N5 is the number of workers to spawn at once. Unsupported in UNIX; specify 0.
NPIPE_NAMES	<pipe_name, pipe_name, ...>	For forking servers on Named Pipes, this specifies the channels the server listens on. A comma-separated list of Named Pipes pipe names.
REUSE_DB_CONNECTION	Y N (default)	For banking servers, this keeps the connection between the server and database open between client requests. The database connection is closed when the client context closes. Relevant only if [DNA] section LOG_UNIT_OF_WORK setting is set to NONCONVERSTIONAL.

RTX_EXITDATALEN		This setting determines the number of bytes in the exit-data field required by the Routing and Transaction Switching exit. As the size of a CICS COMMAREA cannot exceed 32K, the listener will reduce the exit-data length accordingly so the RTXCA fields would not exceed 32K. That the CICS recommended maximum COMMAREA length is 25K. The data in the exit-data field are maintained across invocations of the Routing and Transaction Switching exit program. This is specified in the DNA_EXITS section, via the SRV_RTX_EXIT variable. Due to the nature of CICS Web Support, the contents of the buffer will NOT be maintained across calls to the Routing and Transaction-Switching Exit in this environment. This does not cause any problems or restrictions on the supplied DNARTX. Existing user exits (for the TCP Listener) may rely on the buffer contents and thus would need to be modified for CICS Web Support usage.
SERVER_DNARPC_TIMEOUT	600 (default)	For servers, this specifies the number of seconds without communication from a client. The popping of this timer returns control to the client with an error number of 455. If the variable is set to 0, no timer is activated.
SERVER_POLLING_INTERVAL	250 (default)	For servers, this controls the number of milliseconds to wait before polling for data. A value of 0 is equivalent to the default. Values less than 100 milliseconds may adversely affect other executing processes. Supported for LU6.2 and Named Pipes connections only.
SERVER_RECV_TIMEOUT	<integer value>	If after receiving a connection request but before the specified value (in seconds) expires, and the listener has not received the required minimum amount of data, the listener will assume the connection is suffering from a communications problem (for example, DoS attack). The listener will issue an error message and close the socket. During the time-out period, the Listener will probably consume CPU cycles, as there is always "data" waiting to be processed. However, other connections, existing and new requests, will be processed.
SERVER_SHUTDOWN_TIMEOUT	5 (default)	For workstation servers, this specifies the amount of time in seconds to wait after the user invokes the server shutdown command before the server and its child processes are shut down.
SERVICES_PER_SERVER	default: 100 (for UNIX) 50 (for other platforms)	For workstation servers, this specifies the number of services the server can invoke per connection. The number include the service called by the client and any services the application subsequently calls. The longer the connection stays open (as specified in the [DNA] section LOG_UNIT_OF_WORK setting), the greater the number of services the server may invoke. Do not set this value too conservatively for your application. The system memory allocated per service at initialization is minimal.

SRV_CLOSE_SEMANTICS	COMMIT (default) ABORT	For servers, this specifies how to handle the commit as specified in the [DNA] section LOG_UNIT_OF_WORK setting. **COMMIT** - Use if you want the server to commit the transaction before replying to the client. **ABORT** - Use if you want the server to commit the transaction only if the reply to the client succeeds. ⚠ This setting is irrelevant when the [DNA] section LOG_UNIT_OF_WORK setting is set to CONVERSATIONAL.
SRV_DLL_DIR	{install_dir}\nt\sys\bin, {install_dir}\nt\rt\SERVER	For servers, this specifies a comma-separated list of paths to subdirectories that hold services DLLs.
TCP_PORT_NUMBER	<port_number>	This setting is used for MVS only. It is strongly recommended to specify the port number in the EZAC configuration record. However, if none exists, the Listener may attempt to use the value specified for the TCP_PORT_NUMBER variable.
TCPIP_ABENDPROGRAM	<abend handler program name>	This setting is used for MVS only. This setting specifies the name of a user-supplied abend handler program. If a value is provided, the Listener issues a 'HANDLE ABEND PROGRAM(abhndler)'. If the listener abends, this program can take the necessary steps to recover. A sample program, DNAABEND, is provided in both source (C Language) and object format. As provided, this handles the Workload Management (WLM) de-registration. However, you can add functionality to suit your needs, for example, to restart the Listener.
TCPIP_CLOSEWAIT	blank <positive number> <negative number> 0	This setting is used for MVS only. This setting specifies the timeout value in seconds. This setting provides control over session termination processing. After sending the last response to the client, the CICS service (AppBuilder Communications) closes the connection. This can be done in any of the following ways: Option 1: Issue a CLOSE() for the socket. Option 2: Issue a SELECT with a timeout as specified for TCPIP_CLOSEWAIT, and after either receiving data (the close indication) from the client or when the timeout expires, issue a CLOSE() for the socket. Option 3: Issue a blocking RECV() and after receiving data (the close indication) from the client, issue a CLOSE() for the socket. This behavior is determined primarily according to the value of TCPIP_CLOSEWAIT: If the TCPIP_CLOSEWAIT timeout is not specified or is empty - Do NOTHING (Option 1). If the TCPIP_CLOSEWAIT timeout is a positive number - Issue a SELECT (Option 2). If the TCPIP_CLOSEWAIT timeout is invalid, negative or zero - Issue a blocking RECV (Option 3). TCP Listener checks this value when the Listener detects an error. If a positive numeric value is specified, the Listener does not terminate the session immediately. Instead, it waits for the workstation to receive the response data, and the workstation initiates the session termination. While the workstation is terminating the session, the Listener can processes other requests. In addition, if the problem is due to either the workstation or the network, the session termination does not arrive at the host. The Listener closes the connection after the timeout value specified in the ACCTIME parameter of the EZAC configuration record.
TCPIP_LINGERTIME	0 <positive number>	This setting is used for MVS only. This specifies the number of seconds after the connection is closed, the data sent is still considered valid (SO_LINGER). If an invalid value is specified, SO_LINGER is set to the default value (5 seconds). To disable the Linger option, specify a LingerTime of 0 (zero); SO_LINGER will not be set.

TCPIP_PEEKBUFSIZE	144 (default)	This setting is used for MVS only. This specifies the size of the data buffer passed by the TCP/IP Listener to the user exit's decode routine (dna_TcpMVSDecode). The user requires (header) data in the decoded, uncompressed format, for example to determine the UserID/Password and Rule's short-name. Based on the encoding/compressing algorithm, the amount of data required varies. The default value is 144, which is also the minimum. If a lower value is specified, a value of 144 is used.
TCPIP_SERVICES	<service_name, service_name, ...>	For forking server on TCP/IP, this specifies the channels the server listens on. The value is a comma-separated list of TCP/IP service names.
USE_IMP_NAMES	Y N (default)	For servers, this specifies whether or not the server inspects an implementation file to determine the DLL to be invoked in response to a service request. • Y: The IMPNAME_FILE setting must be set. • N: The server invokes a DLL whose name is identical to the service name.
USE_PREAMBLE_RULEID	Y N	This setting is used for MVS only. When set to Y, it indicates to the listener that the rule's short-name must be taken from the preamble data. This is typically used when implementing security cookies, which are not handled by the listener. ⚠ There are circumstances in which the rule ID in the preamble will be in ASCII. If the name is determined to be in ASCII (first character in the range x'41'-x'5A' (ASCII A-Z)), it is converted to EBCDIC. This conversion includes characters, lower and upper case, digits and blank: ABCDEFGHIJKLMNOPQRSTUVWXYZ, abcdefghijklmnopqrstuvwxyz, 0123456789 and ' ' (blank).
USE_SECURITY_RC	YES NO (default)	If USE_SECURITY_RC is set to 'YES', the listener will provide a return code detailing the nature of the security failure. A return code of 406 indicates a security error. • 702: Invalid user ID • 703: Invalid password • 704: Expired password • 705: Revoked user ID

IMPLICIT_EVENTS

The settings in [IMPLICIT_EVENTS] section are described in the table below.

Dna.ini [IMPLICIT_EVENTS] section

Key	Possible Values / Default	Description
EVENT_SERVICE	EVENT_TABLES (default) SERVICE_MANAGER EXTERNAL	For clients, this specifies the source of information for implicit (table-driven) eventing. **EVENT_TABLES** - used for event tables **SERVICE_MANAGER** - used for AppBuilder Communications Service Manager **EXTERNAL** - used for third-party directory service such as DCE Cell Directory Service
EVENT_STACKS		For clients, use when the EVENT_SERVICE setting is set to EVENT_TABLES. This setting specifies the path and name of the triggers file.
EVENTS_FILE	2 (default)	For clients, use when the EVENT_SERVICE setting is set to EVENT_TABLES. This setting specifies the number of recursive loops executed when a service is a trigger on one platform and an action on another.
TRIGGERS_FILE		For clients, use when the EVENT_SERVICE setting is set to EVENT_TABLES. This setting specifies the path and name of the events file.

LU2

The settings in [LU2] section are described in the table below.

Dna.ini [LU2] section

Key	Possible Values / Default	Description
LOGOFF_SCRIPT	MFLOGOFF.SCR	For clients on LU2, specifies the name of the logoff script file. If the variable is missing or empty, the name is MFLOGOFF.SCR. The SCRIPT_DIR setting identifies the path.
LOGON_SCRIPT	MFLOGON.SCR	For clients on LU2, this specifies the name of the logon script file. If the variable is missing or empty, the name is MFLOGON.SCR. The SCRIPT_DIR setting identifies the path.
LU2_DLL	PCSHLL32 (IBM Personal Communications) EHLAPI32 (NetManage Rumba) ACS3EHAP (Attachmate EXTRA!)	For clients on LU2, this specifies the EHLLAPI DLL name for the terminal emulator in use. Do not specify the .DLL extension.
SCRIPT_DIR	{install_dir}\nt\rt\SCRIPT	For clients on LU2, specifies the path to the LU2 logon and logoff scripts. See also LOGON_SCRIPT and LOGOFF_SCRIPT settings.

LU6.2

The settings in [LU6.2] section are described in the table below.

Dna.ini [LU6.2] section

Key	Possible Values / Default	Description
MODE_NAME	<APPC mode name>	For clients on LU6.2, this specifies the APPC mode name used to contact the remote system.

MESSAGING

The settings in [MESSAGING] section are described in the table below.

Dna.ini [MESSAGING] section

Key	Possible Values / Default	Description
IPC_KEY	NetEssen 30 Msg	For clients and agents for native messaging, this specifies a comma-separated pair of values specifying named shared memory and named semaphores. Only the shared memory value is required for clients.
MEMORY_CONFIG	10:6:-1, 16:19, 10, 5 (UNIX) 10:6:-1, 16:19, 10, 10 (Windows)	This is for system use only. Do not modify without consulting Customer Support.

MVS_LISTENER

The settings in [MVS_LISTENER] section are described in the table below.

Dna.ini [MVS_LISTENER] section

Key	Possible Values / Default	Description

LISTENER_SECURITY	NO PART YES (or CICS) EXIT EXITONLY AUTHENTX AUTHENTXONLY AUTHORX AUTHORXONLY	There are two aspects to this setting. The first relates to security authentication/verification and the second relates to whether to "pass on" the supplied user ID. Security Authentication: **NO** - No security verification is done. **PART** - No security verification is done. The listener's user ID will be associated with the rule transaction. **YES (or CICS)** - EXEC CICS VERIFY command will be issued to verify the user ID and password. If successful, the provided user ID will be associated with the rule transaction. Errors will result in message 934 (and a code of 406 returned to the client). **EXIT and AUTHENTX** - The Authentication Exit is called to verify the user ID and password. If successful, the provided user ID will be associated with the rule transaction. **EXITONLY and AUTHENTXONLY** - This will cause the listener to invoke the user's Authentication Exit to verify the user ID and password. If successful, the listener's user ID will be associated with the rule transaction. A non-zero return code will result in message 301 (and a code of 406 returned to the client). **AUTHORX** - The Authorization Exit is called to verify the user ID and password. If successful, the provided user ID will be associated wit the rule transaction. **AUTHORXONLY** - The Authorization Exit is called to verify the user ID and password. If successful, the listener's user ID will be associated with the rule transaction. User-ID: There are two "types" of EXEC CICS START commands; one with the user-ID and one without. When specifying 'YES' (or 'CICS'), 'PART' or 'EXIT', the user-ID will be used. With the 'NO' and 'EXITONLY' settings, no user-ID is used (and the started transaction will "inherit" the user-ID associated with the listener task.) Regardless of whether the user-ID was used or not, any non-zero return code from the EXEC CICS START command will result in error message 414. This is a generic EXEC CICS xxx error message, not specific to failed EXEC CICS START commands. This option is not available for UNIX.

NLS

The settings in [NLS] section are described in the table below.

Dna.ini [NLS] section

Key	Possible Values / Default	Description

DATE_FORMAT	<standard AppBuilder formats, except for the formats that specify words, such as 'ONE', 'JANUARY', 'MONDAY', etc.>	This setting is used for MVS only, and applies to OpenCOBOL only. This setting specifies the format that AppBuilder Communications is to use when converting DATE type fields exchanged between the client and server. *Token descriptions:* **%m** - Month, numeric (1...12), example: 2 **%0m** - Month, numeric, with leading zero (01...12), example: 02 **%d** - Day, numeric (1...31), example: 28 **%0d** - Day, numeric, with leading zero (01...31, example: 28 **%c** - Year, numeric, first 2 digits (century -1), example: 19 **%0c** - Year, numeric, first 2 digits with leading zero (century -1), example: 9 **%y** - Year, numeric, last 2 digits (00...99), with the first two digits implied to be 19, example: 95 **%0y** - Year, numeric, last 2 digits, with leading zero, examples: 95 **%Y** - Year, numeric, all 4 digits (0000...9999), example: 1995 *Certain shorthands are also accepted:* **International Standards** Key: ISO Date format: yyyy-mm-dd, example: 1987-10-12 Word: Organization Time: hh.mm.ss, example: 13.30.05 **IBM USA standard** Key: USA Date format: mm/dd/yyyy, example: 10/12/1987 Time:hh:mm AM/PM, example: 1:30 PM **IBM European standard** Key: EUR Date format: dd.mm.yyyy, example: 12.10.1987 Time: hh.mm.ss, example: 13.30.05 **Japan Industrial Standard** Key: JIS Date format: yyyy-mm-dd, example: 1987-10-12 Time: hh:mm:ss, example: 13:30:05  This setting is in the [OPEN-COBOL] section of the dna.ini on the mainframe.
EIB_COMMAREA	N	This setting is used for MVS only, and applies to OpenCOBOL only. This setting specifies whether or not to pass EIB and COMMAREA (addresses) as parameters to the invoked rule. Currently the only supported value is 'N'.  This setting is in the [OPEN-COBOL] section of the dna.ini on the mainframe.
ENABLED	Y N (default)	This setting is used for MVS only, and applies to OpenCOBOL only. This setting specifies whether or not this is an OpenCOBOL environment.  This setting is in the [OPEN-COBOL] section of the dna.ini on the mainframe.

EURO	YES NO	This setting must be manually added and set to "YES" in order to activate 'Euro' symbol support. LOCAL_CODEPAGE in the [NLS] section of the Dna.ini file must be set to cp1252 for the euro symbol to be properly translated. This setting is not used on the mainframe.
INBOUND_EBCDIC_NL_TO_LF	YES NO (default)	This setting specifies the option when converting EBCDIC NL char to LF. When set to YES, it converts NL chars coming from the host into LF chars. See also OUTBOUND_EBCDIC_LF_TO_NL.
LOCAL_CODEPAGE	cp437 (US English) cp819 (iso-8859-1) cp860 (Portuguese) cp863 (French) cp865 (Nordic) cp874 (Thai) cp932 (Shift_JIS) cp949 (Korean) cp1004 (Windows Latin 1) cp1252 (Windows Latin1) cp1253 (Greek with Euro) ibm-037 (US English - EBCDIC) ibm-273 (German - EBCDIC) ibm-277 (Danish - EBCDIC) ibm-278 (Swedish - EBCDIC) ibm-280 (Italian - EBCDIC) ibm-284 (Spanish - EBCDIC) ibm-285 (English Ireland - EBCDIC) ibm-297 (French - EBCDIC) ibm-500 (International latin1 EBCDIC) ibm-875 (Greek EBCDIC) ibm-930 (Japanese EBCDIC) ibm-933 (Korean EBCDIC) ibm-939 (Kanji EBCDIC) ibm-1047 (Open Systems Latin 1 EBCDIC)	For clients and servers, this specifies the codepage used in the local environment. If this value is different from the value set in the DEFAULT_CODEPAGE setting in the [ROUTING] section, a codepage translation occurs. The data is converted to the codepage set in the DEFAULT_CODEPAGE setting of the [ROUTING] section. Refer to the *Communications Guide* for more information about the codepage.
OUTBOUND_EBCDIC_LF_TO_NL	YES NO (default)	This setting specifies the option when converting EBCDIC NL char to LF. When set to YES, it converts LF chars going to the host into NL chars. See also INBOUND_EBCDIC_NL_TO_LF.
TIME_FORMAT	%0t.%0m.%0s.%0f	This setting is used for MVS only, and applies to OpenCOBOL only. This setting specifies the format that AppBuilder Communications is to use when converting TIME type fields exchanged between the client and server. Currently the only supported format is %0t.%0m.%0s.%0f (CodeGen restriction) ⚠ This setting is in the [OPEN-COBOL] section of the dna.ini on the mainframe.

ROUTING

The settings in [ROUTING] section are described in the table below.

Dna.ini [ROUTING] section

Key	Possible Values / Default	Description
DEFAULT_CODEPAGE	<codepage> cp1004 cp932	For clients, this setting must be specified only when NAME_SERVICE setting is set to DNAINI. This setting specifies the codepage on the server machine. If the codepage is specified, then the conversion is performed locally. Otherwise, the conversion is performed on the server machine.
DEFAULT_HOST_NAME		For clients, this setting must be specified only when NAME_SERVICE setting is set to DNAINI. This setting specifies the host name of the server machine as known to the protocol for the route (or CICS region name for CICS servers).
DEFAULT_PROTOCOL	lu2 lu62 tcpip	For clients, this setting must be specified only when NAME_SERVICE setting is set to DNAINI. This setting specifies the protocol that is used to connect to the server.
DEFAULT_SERVERID	PCIO (for CICS) ne20 (for PC or UNIX server)	For clients, this setting must be specified only when NAME_SERVICE setting is set to DNAINI. This setting specifies the server ID.
DNA_MAX_ROUTES	<numeric value> 1 (default)	For clients, this specifies the number of routes to be tried for an RPC. If NAME_SERVICE setting is set to DNAINI, this setting must be 1. If NAME_SERVICE=ROUTE_TABLE, then this setting must be equal to the number of servers in the topology.
DNA_NO_OF_RETRIES	<numeric value> 0 (default)	For clients, this setting specifies the number of times the AppBuilder Communications retries in case of failed calls. Routes failures typically result from network problems or missing or corrupted services. The return of an error code from a service does not constitute a route failure.
DNA_RETRY_INTERVAL	<numeric value (seconds)> 0 (default)	For clients, this specifies the number of seconds between the failure of a service route and a retry. The interval increments by a factor of 2. When set to 10 seconds, it increments to 20 seconds after the first retry, 40 seconds after the second retry.
NAME_SERVICE	DNAINI (default) ROUTE_TABLE SERVICE_MANAGER EXTERNAL	For clients, this specifies the source of routing information. **DNAINI** - Use if routing information is to be taken from the DNA.INI. **ROUTE_TABLE** - Use if routing information is to be taken from the routing table. **SERVICE_MANAGER** - Use if routing information is to be taken from AppBuilder Communications Service Manager. **EXTERNAL** - Use if routing information is to be taken from a third-party directory service such as DCE Cell Directory Service.
ROUTETBL	{install_dir}\nt\rt\RTABLE	For clients, it is used only when NAME_SERVICE setting is set to DNAINI. This setting specifies the path and file name of the route table.

SMA

The settings in [SMA] section are described in the table below.

Dna.ini [SMA] section

Key	Possible Values / Default	Description
GRE	{install_dir}\nt\sys\bin\grebstnt.dll	It is for internal use and should not be changed by the users.
HPS	{install_dir}\nt\sys\bin\dna2mstr.dll	It is for internal use and should not be changed by the users.
PC_KEY		For UNIX clients for global eventing. This is the IPC key that uniquely identifies a shared memory segment. The value must be the same for all DNA.INI instances of the host.
MAX_SHMEM_SEGS	100 (default)	For UNIX clients, this specifies the number of shared memory segments available to AppBuilder for global eventing.
SMA_HOST_NAME	<name of the workstation>	For clients and agents for global eventing, this specifies the protocol-independent name of the current machine.
SMA_MVS_TRAN	default: DEVT (for CICS) DNAIEVNT (for IMS)	For MVS servers for global eventing, this specifies the transaction used to run the eventing program. For CICS, the default is DEVT, which starts the program DNACEVNT. For IMS, the default is DNAIEVNT, which schedules a program of the same name.

SMA_NAME	<name of the workstation>	This is for system use only. Do not modify.
SMA_SMSA_CODEPAGE	cp1004 cp932	For agents for global eventing, this specifies the codepage in use on the cell manager machine. This is equivalent to he route table Codepage field.
SMA_SMSA_HOST_NAME	<name of the workstation>	For clients for global eventing, this specifies the protocol-independent name of the current machine. For agents for global eventing, this specifies the protocol-independent name of the cell manager machine. In both cases, this is equivalent to the route table Host field.
SMA_SMSA_NAME	<name of the workstation>	For clients for global eventing, this specifies the name of the current machine. For agents for global eventing, this specifies the name of the cell manager machine. In both cases, this is equivalent to the route table Host field.
SMA_SMSA_PROTOCOL	tcpip	For clients for global eventing, this specifies the network protocol for the connection to the agent. For agents for global eventing, this specifies network protocol for the connection to the cell manager machine. In both cases, this is equivalent to the route table Protocol field. For clients using the SVMD daemon server, specify LOCAL or WINMSG.
SMA_SMSA_SERVERID	ne20	For clients for global eventing, this specifies the server ID of the resident server. For agents for global eventing, this specifies the server ID of the resident server on the cell manager machine. In both cases, this is equivalent to the route table Server ID field.
SMA_SUBCELL_TABLE	{install_dir}\nt\sa\subcell.tbl	For cell managers for global eventing, this specifies the path to the subcell table. Set this variable to EXTERNAL if the NAME_SERVICE setting is set to EXTERNAL in the [ROUTING] section.
<*SUBSYSTEM_NAME*>		For clients for global eventing, the variable name is the name of a subsystem. The value is the path and file name of a notification exit of that subsystem. The subsystem name is identified in the subsystem argument of the dna_RegisterEvent function. A notification exit is not required in the AppBuilder environment.

TRACING

The settings in [TRACING] section are described in the table below.

Dna.ini [TRACING] section

Key	Possible Values / Default	Description
BACKUP_FILE_NAME	{install_dir}\trace.bak	For clients and servers, this specifies the path and name of the backup file that tracing information is written to when the size of the file specified in the TRCFILE setting exceeds the value specified in the TRACE_FILE_SIZE setting. The new information overwrites the existing information in the file.

DEBUGLVL	ERRORS (or 0) (default) ERRORS_AND_LENGTHS (or 1) ERRORS_AND_DATA (or 2) ERRORS_AND_TRACES (or 3)	For clients and servers, this determines what is written to the file specified in the TRCFILE setting. **ERRORS (or 0)** - This logs errors. **ERRORS_AND_LENGTHS (or 1)** - This logs errors, the time the unsuccessful function was called, the time the function exited, and the input/output view lengths. **ERRORS_AND_DATA (or 2)** - This logs errors, the time the unsuccessful function was called, the time the function exited, the input/output view lengths, and a hex dump of the view data with the type of view to be written and the number of bytes to be written per view determined by the values set in the DUMPDATA setting, DUMP_INPUT_SIZE setting, and DUMP_OUTPUT_SIZE setting. **ERRORS_AND_TRACES (or 3)** - This logs errors, the time the unsuccessful function was called, the time the function exited, the input/output view lengths, a hex dump of the view data with the type of view to be written and the number of bytes to be written per view determined by the values set in the DUMPDATA setting, DUMP_INPUT_SIZE setting, and DUMP_OUTPUT_SIZE setting, and trace information for successful calls. For both successful and unsuccessful calls, AppBuilder provides a hex dump of the input and output view with the number of bytes determined by the [TRACING] settings listed above. For unsuccessful calls, AppBuilder also logs status view data.  Use ERRORS_AND_TRACES only when debugging AppBuilder applications. It can degrade system performance and use excessive disk space.
DUMP_INPUT_SIZE	512 (default)	For clients and servers, use when DEBUGLVL = 2 and DUMPDATA is set to INPUT or INOUT, or when DEBUGLVL = 3. This determines the number of bytes per input view to be written the file specified in TRCFILE. The complete input view is written if the value is greater than the number of bytes in the view.
DUMP_OUTPUT_SIZE	512 (default)	For clients and servers, use when DEBUGLVL = 2 and DUMPDATA is set to OUTPUT or INOUT, or when DEBUGLVL = 3. This determines the number of bytes per output view to be written the file specified in TRCFILE. The complete output view is written if the value is greater than the number of bytes in the view.
DUMPDATA	INPUT OUTPUT INOUT	For clients and servers, use when DEBUGLVL = 2 to determine the type of view data written to the file specified in TRCFILE. **INPUT** - for input views **OUTPUT** - for output views **INOUT** - for both To specify how much data is to be written, set the DUMP_INPUT_SIZE and/or DUMP_OUTPUT_SIZE.
HPSCATS	0:err_jp.cat, 8:hdb.cat	For system use only. Do not modify the value. For clients and servers, this specifies a colon-separated list of language-specific message catalogues.
HPSCATSFILES	{install_dir}\MSG	For system use only. Do not modify the value. For clients and servers, this specifies the path to the directory containing the message catalogues. The LANG setting identifies language-specific subdirectories.
LANG	C	For system use only. Do not modify the value. For clients and servers, this specifies the path to the directory containing the message catalogues. The HPSCATSFILES setting identifies the path to the subdirectory.
TRACE_FILE_SIZE	1M (default)	For clients and servers, this specifies the number of bytes of tracing information to be logged to the file identified by the TRCFILE setting before the system deletes the information and stores it in the backup trace file specified in the BACKUP_FILE_NAME setting. Specify an integer followed by a K (for kilobytes) or an M (for megabytes).

TRCFILE	dnatrace.out (default)	For clients and servers, this specifies the path and name of the file to which AppBuilder writes tracing information. Trace output for the mainframe goes to C/370 standard output. Errors that occur before initialization are reported in dnatrace.out on the PC, and in /tmp/dnatrace.out on UNIX.

Appbuilder ini Settings

Appbuilder.ini Settings

The Appbuilder.ini file contains initialization information on the operation and configuration of Java applications. This ini file as well as the Font.ini file (see Font.ini Settings) must be properly set in order to execute AppBuilder Java applications.

The Appbuilder.ini file is located in the java\rt folder under the AppBuilder install directory. The settings in the AppBuilder.ini file affect the following type of applications:

- Java Client
- HTML Client
- EJB
- SOAP (Simple Object Access Protocol) Web Services
- Java Reports

Sections in the Appbuilder.ini

Sections in the Appbuilder.ini

The Appbuilder.ini file contains the following sections.

[BATCH]	[NC]	[VALIDATION]
[CCOMPATIBILITY]	[REPORTWRITER]	[WEBSERVICE]
[DB]	[SERVLET]	
[DEBUG]	[TEST]	

{install_dir} in the Possible Value/Default column indicates the AppBuilder installation directory.

BATCH

[BATCH]

This section has one setting which is used to define attributes for the Java "batch" runtime environment. It specifies the name of the user exit which will be invoked when a "batch" rule is executed from the command line. The user supplied class can establish or change the "context" of the batch modules before and after any batch root rule is executed.

Table 4-1 Appbuilder.ini [BATCH] section

Key	Possible Values / Default	Description
JBATCH_EXEC_EXIT_CLASS		It is a user developed class name that implements the AbfBatchExecExit interface. If this class is contained within a package name, it will be packagename.classname and the compiled class file should be placed in a location accessible through the classpath.

CCOMPATIBILITY

[CCOMPATIBILITY]

The settings in [CCOMPATIBILITY] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. This section contains settings to support features and behavior in the pre Seer*HPS 5.4.1 environments that are deprecated or modified in the AppBuilder releases. These settings could be used to get the old behavior within AppBuilder.

Table 4-2 AppBuilder.ini [CCOMPATIBILITY] section

Key	Possible Values / Default	Description
BYTE_ORDER	LITTLE_ENDIAN (default) BIG_ENDIAN	This setting specifies the byte ordering used when performing overlay operations involving integer fields for compatibility with native code, for example, C Client rules. It must be set according to the CPU architecture. For PCs this must be set to LITTLE_ENDIAN.
MENUBAR_HEIGHT_ADJUSTMENT	<numeric value> 0 (default)	This setting specifies the menu bar height adjustment in pixels to compensate for the height difference between JAVA & C (Windows) menus. Negative value means decrease the height by the number of pixels specified, and positive value means increase height by the number of pixels. For Seer*HPS 5.4.1 size compatibility, this can be set to -4.
STRING_FORMAT_CHANGE_DATA	TRUE FALSE (default)	This setting is for backward compatibility. When set to TRUE, the string display format changes the data. Seer*HPS 5.4.1 and earlier versions used to change the case (UPPER/LOWER) of the character data entered according to the display format. In AppBuilder, the display format is used only to format the data for displaying and the edit mask must be used to filter the inputs to a defined alpha case. By setting this flag to TRUE, the Seer*HPS 5.4.1 behavior is used and the STRING FORMAT is used to store the data.
TOGGLE_CHECK_MENU_ONCLICK	TRUE FALSE (default)	Clicking on a MenuItemCheckBox will toggle the selection depending on this flag. When set to TRUE, any menu items that are 'checkable' will be automatically toggled when clicked. An item is flagged 'checkable' if it's painted as checked in Window Painter, or its state is explicitly set at runtime by the application, whether checked or unchecked.

_DB_

[DB]

The settings in [DB] section are used for the Java Client, HTML Client, SOAP Web Services and Java Reports. The settings in this section control the database access locally, as well as the creation of the transaction context. [DB] is the default section name to access a database. When it is necessary to define multiple data sources, create DB sections as in [DB.dsn] format (dsn is the name of the data source), and set the necessary parameters.

> ⚠ EJB's get the database information from the APPSERVER.

Table 4-3 AppBuilder.ini [DB] section

Key	Possible Values / Default	Description
DB_ACCESS	LOCAL APPSERVER	This setting specifies where to access the database. • LOCAL: to access database locally • APPSERVER: to access database from application server
DBMS_PREFIX	jdbc:db2: (default)	This setting specifies the driver prefix (used for LOCAL access).
DBNAME	<database name>	This setting specifies the database name to connect to (used for LOCAL access). When accessing a database using Rules generated using a Server Partition, then you can use an exit routine defined in a server side INI file known as AppBuilderCOM.INI
JDBC_DRIVER	default: COM.ibm.db2.jdbc.app.DB2 Driver	This setting specifies the jdbc driver name (used for LOCAL access).

INITIAL_CONTEXT_FACTORY	<java_class_name>	These settings specify the initial context factory and provider URL. These are used to access an initial context for an application server to handle transactions and database connections. This is applicable only for DB_ACCESS=APPSERVER and the following are the only supported environments (WebLogic and WebSphere) for DB_ACCESS=APPSERVER setting. Sample entries for BEA WebLogic AppServer: INITIAL_CONTEXT_FACTORY=weblogic.jndi.WLInitialContextFactory PROVIDER_URL=t3://localhost:7001 Sample entries for IBM WebSphere 3.5 AppServer. INITIAL_CONTEXT_FACTORY=com.ibm.ejs.ns.jndi.CNInitialContextFactory PROVIDER_URL=iiop://localhost:900 Sample entries for IBM WebSphere 4.0 AppServer: INITIAL_CONTEXT_FACTORY=com.ibm.websphere.naming.WsnInitialContextFactory PROVIDER_URL=iiop://localhost:900
PROVIDER_URL	<protocol:// host_name:port_number>	These settings specify the initial context factory and provider URL. These are used to access an initial context for an application server to handle transactions and database connections. This is applicable only for DB_ACCESS=APPSERVER and the following are the only supported environments (WebLogic and WebSphere) for DB_ACCESS=APPSERVER setting. Sample entries for BEA WebLogic AppServer: INITIAL_CONTEXT_FACTORY=weblogic.jndi.WLInitialContextFactory PROVIDER_URL=t3://localhost:7001 Sample entries for IBM WebSphere 3.5 AppServer. INITIAL_CONTEXT_FACTORY=com.ibm.ejs.ns.jndi.CNInitialContextFactory PROVIDER_URL=iiop://localhost:900 Sample entries for IBM WebSphere 4.0 AppServer: INITIAL_CONTEXT_FACTORY=com.ibm.websphere.naming.WsnInitialContextFactory PROVIDER_URL=iiop://localhost:900
PASSWORD		This setting specifies the user password to use when connecting to a database (used for LOCAL access and APPSERVER).
USERID		This setting specifies the userID to use when connecting to a database (used for LOCAL access and APPSERVER).

DEBUG

[DEBUG]

The settings in [DEBUG] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports.

Table 4-4 AppBuilder.ini [DEBUG] section

Key	Possible Values / Default	Description
BROWSER_EXECUTABLE	iexplore.exe (default)	This setting specifies the browser executable filename to run servlets from RuleView. CAUTION: If the full path of the browser executable file contains spaces, please enclose the path in double quotes. Example: "C:\Program Files\Plus!\Microsoft Internet\IEXPLORE.EXE"
CONNECT_TIMEOUT	30 (default)	This setting specifies the time-out value, in seconds, for establishing a connection between the debugger and a remote server (for client/server debugging only).
DBG_JVM_NAME	java.exe (default)	This setting specifies the name of the JVM executable to start in one of the following situations: • when executing the application under RuleView (if NEW_JVM_FROM_MENU=TRUE) • when RuleView is executed on the same JVM as AbfExecutionClient • when RuleView launches a new JVM (its name cannot be changed) to execute a root rule • when the server rule is executed in the debug mode on the RMI server

DBG_JVM_PARAMETERS	-classic	This setting specifies additional parameters that are used when starting the JVM: • from AbfExecutionClient to execute RuleView • from RuleView to execute root rule • on the RMI server to execute server rule in the debug mode Since the debugger is fully supported, use the JVM parameter '-classic' that is executed in the classic VM for JDK 1.3 (HotSpot is used by default). Do not use '-classic' for JDK 1.2 as it is not supported.
RMI_SERVER_HOST	localhost (default)	This setting specifies the host name where the RMI server registry is running. It can be set explicitly even if the registry is running locally.
RMI_SERVER_PORT	1099 (default)	This setting specifies the port number where the RMI server registry is listening. If the registry is listening on a different port, it must be defined here.

RuleView is the AppBuilder's own code debugging tool and requires JPDA (Java Platform Debugger Architecture) on your PC. This is normally included with JDK 1.3 within the TOOLS.JAR file (which must be included in the classpath).

_NC_

[NC]

The settings in [NC] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. These settings control the operation and configuration of Java applications. This functionality was previously called Network Computing (NC).

Table 4-5 AppBuilder.ini [NC] section

Key	Possible Values / Default	Description
ABJ_HOME	{install_dir}\java\rt	This setting specifies the directory runtime locates all the necessary application.
ABORT_ON_COMM_ERROR	TRUE FALSE	This setting specifies whether or (terminate) AppBuilder when a co occurs. In the thin-client environm COMM_ERROR_RULE is run as any other thin-client rule. Therefo ABORT_ON_COMM_ERROR, th USE_COMM_ERROR set to TRU The two settings are mutually exc
ABORT_ON_INVALID _REDEFINE_DATA	TRUE FALSE (default)	This setting specifies the applicat invalid data is put into redefined v to TRUE the runtime exception is application is terminated. When s severe warning is logged and the execution continues. Default valu
AUTH_TYPE	DIALOG EXIT (default) NONE	This setting specifies the Authent used for the following: • when QUERY_AUTHENTICAT is set to TRUE or • when queryAuthenticatio • when a remote AppBuild Possible values are: • DIALOG: Userid/passwo interactively. • EXIT: It invokes the Java in appbuildercom.ini. • NONE: A blank userid/pa all remote calls.

CHAR_COORDINATE_FONT	CCS_FONT (default)	This setting specifies the font that the fontmetrics for char coordinat with the entry defined here must font.ini file.
CLOSE_DETACHED_RULES _WITH_PARENT	TRUE (default) FALSE	This setting specifies whether or rules be closed when the parent it is set to FALSE and the parent the detached rules of the rule are be overridden on an individual rul ParentRuleEndEvent.
COMM_ERR_RULE _PACKAGE		This setting specifies the name of comm_err_rule. This is required COMM_ERR_RULE has a differe from the one used by it's parent r invokes the COMM_ERR_RULE) for the application can be set for Rule, View classes etc., including COMM_ERR_RULE will get the s name as set in the partition.
COMM_INI_URL	<http: //someserver.acompany.com/appbuildercom.ini>	This setting specifies the location appbuildercom.ini file. By default same directory as this file.
DATA_CONVERTER	appbuilder.util.AbfDefaultDataConverter (default) <qualified java class name>	This setting specifies which conv perform redefine overlay and size must be a qualified java class nar This setting is used only if the cor name specified in the rule class c is not a valid converter. Currently appbuilder.util.AbfCOBOLDataCo appbuilder.util.AbfDefaultDataCo
DBCS_ENVIRONMENT	TRUE FALSE (default)	This is for a DBCS enabled versi This value must be set to TRUE i compiled for a DBCS environmen dynamic view mapping optimizati work under a DBCS environment work for an SBCS environment.
DEADLOCK_MONITOR_LOG_PATH		A deadlock monitor is added to th It tracks the running threads and log file for further analisis if a UI f enable the monitor one needs to where the log files will be stored, example:C:\Temp\DLM
DEBUG_NETE_LOAD	TRUE FALSE (default)	This setting specifies whether or debugging of the communications Communications) configuration.
DEFAULT_CENTURY	1900 (default)	This setting specifies the default CHAR(date1, format) function wh the format is '%0y' or '%y'.
DEFAULT_DATE_FORMAT	%0d/%m/%Y	This setting defines the default da used as a display format and an formatting the date values in plac locale specific date format setting format is provided, the current loc is used.
DEFAULT_DOUBLE_FORMAT _HEXADECIMAL	sZZZZZZZZZZZZZZZZZV9ZZZZZZZZZZZZZZZZ E%esZZZ%s	This setting is a default format str function with one HEXADECIMAL
DEFAULT_DOUBLE_FORMAT _IEEE754	sZZZZZZZZZZZZZZZZZV9ZZZZZZZZZZZZZZZZ E%esZZZ%s	This setting is a default format str function with one IEEE754 DOUE
DEFAULT_FLOAT_FORMAT _HEXADECIMAL	SZZZZZZV9ZZZZZE%esZZZ%s	This setting is a default format str function with one HEXADECIMAL

DEFAULT_FLOAT_FORMAT _IEEE754	SZZZZZZZZV9ZZZZZZZE%esZZZ%s	This setting ia a default format str function with one IEEE754 FLOA
DEFAULT_FLOAT _REPRESENTATION_MODE	s c n	This setting allows you to specify used for floating-point numbers c such as s, c, and n. Mode n must integer number and denotes a co mode.
DEFAULT_TIME_FORMAT	%0h:%0m:%0s	This setting defines the default tir used as a display format and an e formatting the time data in place specific time format setting. If no provided, the current locale speci
DEFAULT_TIMESTAMP _FORMAT	%0o-%0d-%Y.%0t.%0m.%0s.%f	This setting defines the default tir used as display format and edit fo formatting timestamp data in plac timestamp format setting. If no de format is provided, the local spec
ENABLE_CLEAR_WINDOW _CHANGES	TRUE FALSE (default)	This setting specifies whether or component CLEAR_WINDOW_C
EXCEPTION_ON _UNSUPPORTED _COMPONENTS	TRUE (default) FALSE	This setting specifies whether or runtime exception on calling unsu components.
FONT_INI_URL	[file:/] {install_dir}\font.ini	This setting specifies the location The Font.ini file must either be re FONT_INI_URL key or must be ir directories included in your CLAS This key is initially set by installat <HPSDIR>.
FRAME_TYPE	WINDOW (default) INTERNAL_FRAME	This setting sets the type of frame application/applet. By default, it is which uses the regular Java Fram AppBuilder windows. If it is set to INTERNAL_FRAME, then the Jav used for the AppBuilder window. is similar to an MDI child window the parent window. Refer to NEW_JVM_FROM_MENU when Frames with an ExecutionClient a application.
HELP_SET_NAME	<filename.ext>	This setting specifies the default I all Java client applications. To us HelpSet, remove the semicolon a and type the actual name of the fi *filename* . Note that calling a Java SetHelpFile() ObjectSpeak metho HELP_SET_NAME entry, for the rules in the hierarchy. For more in setting up help files, refer to the *D Applications Guide* .

IMAGES_DIRECTORY_URL	< http: //someserver.acompany.com/images/>	This setting specifies the location directory. When specifying a list c semicolon (;) as a separator. The contains the jpg files that can be push buttons. By default, the App saves the bitmaps into the image the Java methods look for these images. However, if the ObjectSp setImage or the system compone HPS_SET_BITMAP_FILE is used be under any sub-directory name images. But, the resource files/di right under the current execution setting can be used to put these using a URL. Any given resource given URLs separated by a semi resource is not found in any of th the current execution directory is
INDEX_CONTROL_ABORT	TRUE FALSE	This setting specifies whether or (terminate) AppBuilder when "ind error occurs. If it is set to FALSE, no exception 1 is used for the index.
JVM_NAME	javaw.exe (default)	This setting specifies the name of Machine (JVM) executable to use new JVM from AbfExecutionClien root rule. (That is, when NEW_JV =TRUE.) You may specify the full the path.
JVM_PARAMETERS	-classic	This setting specifies additional p that is started from AbfExecution process root rule.
LOOK_AND_FEEL	com.sun.java.swing.plaf. windows.WindowsLookAndFeel com.sun.java.swing.plaf. motif.MotifLookAndFeel javax.swing.plaf.metal. MetalLookAndFeel	This setting specifies the look-and application. If this setting is not sp does not explicitly set the look-an default. Use only one or comment out.
METAL_COMBOBOX_WORK _AROUND	TRUE	This is a work-around for Java bu (JComboBox does not get focus Metal L&F), and # 4303704 (Meta always reports isFocusTraversab
NEW_JVM_FROM_MENU	TRUE (default) FALSE	This setting specifies whether or launched for every root rule/Rule from the menu. When set to FALS rule/RuleView runs on the same AbfExecutionClient, otherwise a for each root rule/RuleView. If you RuleView, then RuleView launche regardless of this setting. It must enable LOCAL EVENTING (differ running within the same JVM can events). It also must be set to FA Internal Frames (Refer to FRAME the ExecutionClient and the appli the same JVM. When using FALS your CLASSPATH contains the c directory. (AppBuilder\java\rt\loca used).

PROPAGATE_NULL_TO _DATABASE	TRUE FALSE (default)	This setting specifies whether or are propagated to a database. Se property "Null is Allowed", otherw generated. When converting an uninitialized value is set as follows: • In a numeric field, it is se • In a data field, it is set to • In a character field, it is s • In a varchar field, it is set
QUERY_AUTHENTICATION _ON_STARTUP	TRUE FALSE (default)	This setting specifies whether or dialog on startup.
RADIOBUTTON _AUTOSELECT	TRUE FALSE (default)	This setting determines radio butt behaviour on receiving focus. If it is set to TRUE, whenever a ra focus, by whatever means, it is au selected. When set to FALSE, a radio butto specifically selected by the user.
SEPARATE_RPC_THREAD	TRUE FALSE (default)	This setting specifies whether or procedure call (RPC) is run in a s to a Java defect #4096745 (disab AWT [lightweight] components bli window to blink when an RPC is FALSE avoids the blinking, but us program thread to perform the RP making the program (and all othe in the Java virtual machine) unres RPC ends. This Java defect has Sun Microsystems in JDK 1.3.
SHOW_VIEW_WATCH_WIN DOW	TRUE FALSE (default)	This setting specifies whether or watch window when CTRL+F9 is window (shows all the views unde control).
SHOW_ZERO_ON_NULL	TRUE FALSE (default)	This setting specifies whether or NULL data for numeric fields.
USE_COMM_ERR_RULE	TRUE FALSE	In the thin-client environment, the COMM_ERROR_RULE is run as any other thin-client rule. Therefo USE_COMM_ERROR, then you ABORT_ON_COMM_ERROR se versa. The two settings are mutua
USE_SPECIFIED_COLOR_FOR_PROTECTED_FIELDS	TRUE FALSE (default)	This setting can be used to specif for a protected edit field text. If th TRUE, the color specified in the e 'Foreground Color' will be used . If the setting is set to FALSE, the will be used.
WINDOW_ICON	<file_name>	This setting specifies the default i JPEG is the only valid graphic file this file.
YEARS_IN_THE_FUTURE	45 (default)	This setting specifies how dates f is not specified are interpreted.

RuleView is the AppBuilder's own code debugging tool and requires JPDA (Java Platform Debugger Architecture) on your PC. This is normally included with JDK 1.3 within the TOOLS.JAR file (which must be included in the classpath).

REPORTWRITER

[REPORTWRITER]

The settings in [REPORTWRITER] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. This section

contains ReportWriter settings. Refer to the *Reports Guide* for information about creating reports.

Table 4-6 AppBuilder.ini [REPORTWRITER] section

Key	Possible Values / Default	Description
AVERAGE_CHAR_SIZE	20,40 (default)	This setting specifies the average character size (width, height) in the same units as PRINT_POINT_SIZE setting of a particular font in fonts.ini, and is used when ReportWriter can't determine it by other means. This happens when 3800 printer is specified in Report properties and thus all sizes are measured in INCHES/10 instead of characters. In this case ReportWriter can't calculate size of a single character and it will use the value specified here. If sections that are printed horizontally overlap, specify greater value for char width.
DEFAULT_PAGE_SIZE	NA_LETTER (default) ISO_Ax	This setting specifies the default page size. If page setup dialog is not displayed (DISPLAY_SETUP_DIALOG=FALSE), ReportWriter uses this setting as default page size. When specifying ISO_Ax, x is a numeric value ranging from 0 up to 10. All values specified in the class PageAttributes.MediaType can be used here (see JDK 1.3.0 API specification).
DISPLAY_SETUP_DIALOG	TRUE (default) FALSE	This setting specifies whether or not to display the page/printer or other setup dialog. This dialog allows to specify printer name and settings, paper size, etc. If this it is set to FALSE, default values are used for all required settings.
MEDIA	<\\server\printer>	This setting specifies the printer name for PRINTER media type. This setting is required if DISPLAY_SETUP_DIALOG=FALSE
MEDIA_TYPE	PRINTER HTML (not supported) TEXT (not supported)	This setting specifies the media type, which is used to output reports. Currently only PRINTER media is supported. If not specified, a dialog with possible options is displayed upon ReportWriter start.

SERVLET

[SERVLET]

The settings in [SERVLET] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports.

Table 4-7 AppBuilder.ini [SERVELET] section

Key	Possible Values / Default	Description
APPLICATION_TIMEOUT	<millisecond> 300000 (default)	This setting specifies the amount of time (in millisecond) from last access that a session will time-out.
CACHE_TYPE	MEMORY_CACHE FILE_CACHE BOTH (default)	This setting specifies the type of cache. • MEMORY_CACHE: All sessions are retained in memory. • FILE_CACHE: All sessions are saved in the file system of the application server. • BOTH: Sessions are saved partly in memory and the rest in the file system.

CHARACTER_ENCODINGS	<ISO Lang code>=<encoding>
	
	cs=windows-1250;
	en=windows-1252;
	
	ja=shift_jis;
	de=windows-1252
		This setting specifies the encoding to read and write html pages. By default the systems encoding is used to read and display html pages. Use a semicolon (;) to separate multiple entries. You can find a full list of language codes at a number of sites, such as:
		http://www.ics.uci.edu/pub/ietf/http/related/iso639.txt
		
		Java supported encoding can be found at
		http://java.sun.com/j2se/1.4/docs/guide/intl/encoding.doc.html
		
		Character sets in MS IE is found at
		http://msdn.microsoft.com/library/

DISABLE_TOP_AND_BOTTOM_EVENTS	TRUE FALSE (default)	This setting disables LB_TOP and LB_BOTTOM events for Servlet Tables (MCLB). If it is set to TRUE, LB_TOP and LB_BOTTOM events are not generated for MCLB.

ERROR_PAGE	errorpage.html (default)	This setting specifies the name of the HTML file that is used as an error page. When the servlet based application encounters an error, the error code, error description and the context in which the error occurred are displayed as an HTML file. Specify custom error page (with subdirectory if required). The custom error page can optionally include some or all of the above mentioned three error parameters.
MAX_CACHED_SESSIONS	1000 (default)	This setting specifies the maximum number of sessions to have in MEMORY_CACHE. This parameter is valid only when CACHE_TYPE is set to BOTH. All new sessions beyond this limit are saved in FILE_CACHE.
MAX_CONCURRENT_SESSION	10000 (default)	This setting specifies the maximum number of concurrent users. If this number is exceeded, a "Server too busy" error is displayed.
MEMORY_CACHE_TIMEOUT	180000 (default)	This setting specifies the amount of time (in millisecond) from last access that a memory cache. This TIMEOUT is used only when CACHE_TYPE is set to BOTH. Session is transferred to a FILE_CACHE if APPLICATION_TIMEOUT is greater than MEMORY_CACHE_TIMEOUT.

TEST

[TEST]

The settings in [TEST] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. The settings in this section control the operation and configuration of the debugging and testing of an application.

Table 4-8 AppBuilder.ini [TEST] section

Key	Possible Values / Default	Description
DEBUG_START	TRUE FALSE QUERY (default)	This setting specifies whether or not to invoke the RuleView debugger when the main rule starts. When it is set to QUERY, a dialog is displayed, which prompts the user to choose.
DEBUG_URL	<http: //someserver.acompany.com/rulesrc/> default: FILE:/{install_dir}/JAVA/RT/LOCAL/DEBUG (This is in the local classpath.)	This setting specifies the location of the rule source and process files used by the RuleView debugger. If the Partition is used to prepare the application, this needs to be changed to FILE:/{install_dir}/JAVA/RT/ *name of Application Configuration or name of partition* /DEBUG
DISPLAY_LONG_NAMES	TRUE FALSE(default)	This setting has been introduced to control the behaviour of Ruleview. Either the long names or the short names (systemid) of variables in the Watches Data tab can be displayed depending upon the value of the setting.
MENUFILE_URL	<http: //someserver.acompany.com/ menu/menufile.mnu> default: In the local classpath under /menu/ menufile.mnu.	This setting specifies the location of the menu file (menufile.mnu).

RuleView is the AppBuilder's own code debugging tool and requires JPDA (*Java Platform Debugger Architecture*) *on your PC. This is normally included with JDK within the TOOLS.JAR file (which must be inclded in the classpath*).

VALIDATION

[VALIDATION]

The settings in [VALIDATION] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. The settings in this section control the operation and configuration of the validation of a window or field.

Table 4-9 AppBuilder.ini [VALIDATION] section

Key	Possible Values / Default	Description
BEEP_ON_ERROR	TRUE (default) FALSE	This setting specifies whether or not a beep occurs when a field or window validation error occurs.

DBCS_VALIDATION_CODEPAGE	<codepage>	This is for a DBCS enabled version only. This setting specifies the codepage against which the DBCS and MIXED field data will be validated. This is typically the codepage of the ultimate target of such data, for example, the mainframe codepage. Value specified as the Java encoding name for required codepage, for example, CP930 for Japanese mainframe/mixed or CP933 for Korean mainframe/mixed.
DBCS_VALIDATION	USER_INPUT_ONLY NONE ALL	This setting determines the type of validation performed for MIXED and DBCS data. • NONE: No validation is performed. • USER_INPUT_ONLY: Only user's input in MIXED and DBCS fields is validated. • ALL: All actions changing a MIXED or a DBCS field, including rules language statements, are validated. Note: For any validation to be performed, a value must be specified for the DBCS_VALIDATION_CODEPAGE.
FIELD_ERROR_FOREGROUND_COLOR	<valid predefined color constant>	This setting specifies the foreground and background colors for edit fields, multiline edit fields, combo box edit fields, and table cells when a data or format error occurs. The colors must be valid predefined color constants. If no valid colors are specified, the foreground and background text are not altered when an error occurs.
FIELD_ERROR_BACKGROUND_COLOR	<valid predefined color constant> WHITE BLUE	
FIELD_ERROR_TITLE	<title of the error message>	Specify the title for field validation message box This setting has the same meaning as CUSTOM_HELP_TEXT in HPS.INI
FIELD_ERROR_MESSAGE	<text of the error message>	Specify the message for field validation message box This setting has the same meaning as CUSTOM_ERROR_TEXT in HPS.INI
MANDATORY_FIELD_ERROR_TITLE	<title of the error message>	Specify the title for checking mandatory field message box
MANDATORY_FIELD_ERROR_MESSAGE	<title of the error message>	Specify the text for checking mandatory field message box
SHOW_FIELD_ERROR_MESSAGE_BOX_DEFAULT	TRUE FALSE (default)	This setting specifies whether or not an error message box is shown when a field error occurs. By default it is set to FALSE, and the field's FOREGROUND and BACKGROUND color change as per the FIELD_ERROR_FOREGROUND_COLOR and the FIELD_ERROR_BACKGROUND_COLOR settings when there is an error.
SHOW_WINDOW_ERROR_MESSAGE_BOX_DEFAULT	TRUE (default) FALSE	This setting specifies whether or not an error message box is shown when any field in a window has an error when a window action is attempted.
WEAKENED_DATE_FORMAT_VALIDATION	TRUE FALSE (default)	Make the AbfDate format validation weaker, allow '%0Y' and other incorrect (but historically used) formats. DEFAULT: FALSE

WEBSERVICE

[WEBSERVICE]

The settings in [WEBSERVICE] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. The

[WEBSERVICE] section affects how an AppBuilder application executes under web services.

Table 4-10 AppBuilder.ini [WEBSERVICE] section

Key	Possible Values / Default	Description
LOGGING	TRUE FALSE	This setting specifies whether or not to log the XML request and response into a temporary file as a record of all transactions. Each request/response pair can be logged in a temporary file created under the servlet context. Refer to the application server documentation for the exact location of these files. The files have AB as the prefix and .req and .res as extension names.
PARSER_BUF_SIZE	4096 (default)	This setting specifies the buffer size for the parser in bytes. The WebService request handler needs to peek-through the input stream to validate the service name and the request. This peek needs a buffered stream and a buffer size big enough to include this information.
SAX_PARSER	org.apache.xerces.parsers.SAXParser	SAX parser is supplied by various vendors and the package name and Java class name for each one of them will be different. By default, AppBuilder uses the SAXParser from Apache. This can be downloaded from the http://xml.apache.org/xerces-p/index.html. After download, include xerces.jar in the classpath of both client and server. This can be set to other vendor parsers.

Appbuildercom.ini Settings

Appbuildercom.ini Settings

Appbuildercom.ini file contains initialization information on communication when an AppBuilder Java Client calls an AppBuilder Server. The communication settings for a C Client are set in Dna.ini file

Appbuildercom.ini file is located in the java\rt folder under the AppBuilder install directory. The settings in AppBuildercom.ini file affect the following products:

- Java Client
- HTML Client
- EJB
- SOAP Web Services
- Java Reports

Sections in the Appbuildercom.ini

The Appbuildercom.ini file contains the following sections.

[GENERAL]	[ROUTING]
[JAVA_CHAR_ENCODING]	[SERVER.ServiceName]
[NLS]	[TRACING]
[ROUTES]	

[GENERAL]

The settings in [GENERAL] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. These settings specify the general aspects of communications.

AppBuildercom.ini [GENERAL] section

Key	Possible Values / Default	Description
C_BYTE_ORDER	BIG_ENDIAN (default) LITTLE_ENDIAN	By default, Java uses BIG_ENDIAN (ordered from most significant digits to least). C components uses marshalling to convert Java byte stream to a platform dependent byte stream. The BYTE_ORDER has to be set to LITTLE_ENDIAN for using C components in Wintel platforms.

CLOSE_SEMANTICS	COMMIT	
DNARPC_TIMEOUT	300	
LOG_UNIT_OF_WORK	NONCONVERSATIONAL	
OPENCOBOL_DATE_FORMAT	ISO (default)	This setting specifies the date format for OpenCOBOL performance marshalling. Use format names as defined for DB2 (ISO,USA,EUR or JIS) or a format string, using format tokens defined for date formatting in the *Rules Language Reference Guide* .
OPENCOBOL_TIME_FORMAT	ISO (default)	This setting specifies the time format for OpenCOBOL performance marshalling. Use format names as defined for DB2 (ISO,USA,EUR or JIS) or a format string, using format tokens defined for time formatting in the *Rules Language Reference Guide* .
POLLING_INTERVAL	250	
PRODUCT_NUMBER	193	
REFLECTION_ORDER	JAVA_ORDER (default) GENERATED_SEQ	This setting specifies the sequence used for marshalling views and fields. • JAVA_ORDER: equates to usage of standard Java Reflection API • GENERATED_SEQ: guarantees that marshalling occurs in the AppBuilder hierarchy sequence, but is slower
USER_EXITS	UserExitExample	This setting specifies the Java class containing the user exits.
VERSION	3.1	

[JAVA_CHAR_ENCODING]

The settings in [JAVA_CHAR_ENCODING] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. This section contains settings to specify the mapping between the target codepage and Java encoding. The specified encoding is used in the data translation when sending or receiving data to and from the remote server.

AppBuildercom.ini [JAVA_CHAR_ENCODING] section

Key	Possible Values / Default	Description
IBM1004	ISO8859_1	
IBM930	CP930	for Japanese
IBM932	MS932	for Japanese
IBM949	cp949	for Korean
IBM933	cp933	for Korean

[NLS]

The settings in [NLS] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. This section contains settings for the National Language Support.

AppBuildercom.ini [NLS] section

Key	Possible Values / Default	Description
LOCAL_CODEPAGE	IBM1004 IBM932 (for Japanese)	This setting specifies the local codepage when no mapping is defined for server codepage, provided a mapping exists for it.

[ROUTES]

The settings in [ROUTES] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. The [ROUTES] section tells an AppBuilder Java Client how and where to route calls to its server Rules. $ANY setting routes all calls to the same service.
The name of the service you want to use to execute the Rule (or all the Rules) must be defined in its own section [SERVER.ServiceName], where *WebService* is the name of the service, prior to the [ROUTES] section.

AppBuildercom.ini [ROUTES] section

Key	Possible Values / Default	Description
$ANY	nete	This setting specifies the default server entry for all remote rules.

[ROUTING]

The settings in [ROUTING] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. This section contains the settings for routing.

AppBuildercom.ini [ROUTING] section

Key	Possible Values / Default	Description
MAX_ROUTES	2	
NO_OF_RETRIES	0	This setting specifies the number of attempts to connect on failures to a remote server.
RETRY_INTERVAL	0	This setting specifies the time in milliseconds between the reattempts to connect to a remote server.
ROUTES_KEY	LONGNAME SHORTNAME	This setting specifies the name of the rule from which the route table is keyed.

[SERVER. *ServiceName*]

The settings in [SERVER. *ServiceName*] section are used for the EJB, WEBSERVICE, RMI and Custom.
ServiceName contains the name of the service you want to use to execute the Rule (or all the Rules), which must be defined in its own section prior to the [ROUTES] section. For example, to define a service "nete", create a section [SERVICE.nete]. Refer to the *Deploying Applications Guide* for information on using different web services.
See also the [WEBSERVICE] section in the AppBuilder.ini file, which affects how an AppBuilder application executes under web services.

AppBuildercom.ini [SERVER.nete] section

Key	Possible Values / Default	Description
CLASS	CustomTransportExample	
CODEPAGE	IBM1004 IBM932	This setting specifies the codepage of the server for local conversion. For a DBCS-enabled version of AppBuilder, specify IBM930 or IBM939 for Japanese if the server is a mainframe.
HOST_NAME		This setting specifies the HostName of the remote machine.
HOST_URL	=http: //hostname:port/ServletContext/ServletName	
INITIAL_CONTEXT_FACTORY	java_class_name	
PORT	3090	This setting specifies the TCPIP port to connect.
PROTOCOL	TCPIP	This setting specifies the protocol used to connect to the server.
PROVIDER_URL	protocol://host_name:port_number	
REGISTRY_URL	[protocol:]//host_name:port_number	
SERVERID		This setting specifies the server ID of the host server.
TYPE	nete	The type of service.

[TRACING]

The settings in [TRACING] section are used for the Java Client, HTML Client, EJB, SOAP Web Services and Java Reports. This section contains settings for tracing.

AppBuildercom.ini [TRACING] section

Key	Possible Values / Default	Description
DEBUGLVL	TRACES_NONE ERRORS_ONLY ERRORS_AND_DATA ERRORS_AND_TRACES	This setting specifies the level of debug information.
TRCFILE	<file_name> trace.out	This setting specifies the trace file for communication traces. Default is to direct traces to same location as application traces, specified by APP_FILE setting in appbuilder.ini. A file name may be specified with a full or part qualified path. Filenames without any path or part qualified will be created relative to TRACE_DIR and/or environment's temp directory.

Font.ini Settings

Font.ini Settings

The Font.ini file contains font definitions and is located directly under the AppBuilder install directory. The font.ini file externalizes the font mappings used by the client runtime. The client runtime references the font.ini to retrieve the mapping for any logical font names specified in the panel files. The mappings in the font.ini match those previously defined internally to the product. For more information about the font mapping at execution time, refer to the *Development Tools Reference Guide* .

Creating a new Font Definition

In addition to the predefined font definitions, you can also define your own fonts to use in the Windows. To define a font, add a section to the Font.ini file using the font definition syntax as follows:

Font Definition Syntax

Here is a definition of a font:

[*LOGICALNAME*]

platform=face_name, point_size, style1, ..., stylen
GUI_SIZE= *width, height*
PRINT_POINT_SIZE= *width, height*
USAGE= *usage, usage*
where:

Table 6-1 Font definition syntax

Variable	Possible Values / Default	Description
LOGICALNAME		This is a logical name of the font, and is used as the panel file symbol. It must be all uppercase. Punctuation or spaces are not allowed as part of the font name. It must not be all numeric except for fonts used by the Report Writer (see Fonts used by Report Painter)
platform	HTML Java Win32 DotNet	Specify any of the platforms.
face_name		This is the name of the face sent to the underlying operating system. If you are executing the application in the Windows environment, this is the name of the actual font as defined in the Fonts settings for the machine. For example, Arial is not the same as Arial Bold, which is not the same as Arial Bold Italic
point_size		This is the positive integer size sent to the operating system.

style1, ... stylen	BOLD ITALIC UNDERSCORE STRIKEOUT	UNDERSCORE and STRIKEOUT are not supported for Java. When specifying more than one style, separate them with commas. Not all styles are supported by all platforms
GUI_SIZE= *width, height*		This setting is empty by default. This setting overrides the font size reported by the operating system. Width and height are positive integer numbers, specified as points or pixels for resolution of 240 dpi, which means 1 point equals 1/240 of an inch. This does not include edit field borders.
PRINT_POINT_SIZE= *width, height*		This setting specifies the point size for Report Painter. Width and height are positive integer numbers, specified as points or pixels for resolution of 240 dpi, which means 1 point equals 1/240 of an inch.
USAGE= *usage*	Window Report	This setting specifies which tool uses this font. • Window: indicates Window Painter • Report: indicates Report Painter and Report Writer When specifying more than one usage, use a comma to separate the values.

The following is an example of the font definition to create a new font called MYFONT:

[MYFONT]

Java=monospaced,13
Win32=Arial,10
DotNet=Arial,10
HTML=Arial,10
Usage=Window
The new font is available to use in the Windows the next time you start the AppBuilder.
When you run a Java or J2EE application written in the AppBuilder, the new font definition you have created must also be included in the Font.ini file in the execution environment. The location of the file depends on the setting in the AppBuilder.ini file.
In order for Java to handle characters correctly, specify a logical font name in the font.ini and add a mapping for that logical name in the font.properties (font.properties.ja for Japanese setup, and font.properties.ko for Korean setup).

Example: Font mapping for DBCS characters

Define the following font in the font.ini file:

[MSGOTHIC8]

Win32=MS Gothic,8
DotNet=Arial,8
Java=msgothic,11
Usage=Window

Locate 'font.properties.ja' and add the following:

msgothic.plain.0=MS Gothic,ANSI_CHARSET
msgothic.plain.1=MS Gothic,SHIFTJIS_CHARSET
fontcharset.msgothic.0=sun.io.CharToByteCp1252
fontcharset.msgothic.1=sun.io.CharToByteMS932
exclusion.msgothic.plain.0=0100-20ab,20ad-ffff

Sections in the Font.ini

The section name in the Font.ini file represents a font name and the entry lines within the section defines the font. The following is a list of predefined font definitions:

- Modern8, 10
- Swiss8, 10, 12, 14, 18, 24
- Roman8, 10, 12, 14, 18, 24
- SystemFont8

The following sections in the Font.ini file contain font information used for special purposes:

- [CCS_FONT]
- [CRULEVIEW]
- Fonts used by Report Painter

[CCS_FONT]

This font is used to create the fontmetrics for Char Coordinate System.

[CCS_FONT]

Java=sansserif,12,BOLD
Win32=System,10
HTML=System,10
Usage=Window

For C#.NET applications, if the DotNet entry is not set, the WIN32 value is used, in order to have window sizes close to C or Java applications. Depending on other machine settings, it may have to be changed to have the following value:

Win32=System,8

[CRULEVIEW]

This font is used by C RuleView.

[CRULEVIEW]

Win32=Microsoft Sans Serif,10

Fonts used by Report Painter

The font definitions [001] through [024] are used by the Report Painter for mainframe reports. The Report Painter stores font numbers in the sections files. To resolve fonts at runtime, the Report Writer must be able to find a font by its number. All fonts that are used in reports must have <logical name> equal to a font number.
For example,

[001]

DisplayName=Gothic Text 10-pitch
Java=Courier,12,bold
Win32=Courier,12,bold
PRINT_POINT_SIZE=24,40
Usage=Report

Appcfgdefault.ini Settings

AppBuilder allows the code to be generated on a workstation in preparation to remotely preparing objects to a mainframe using a Workgroup repository rather than a mainframe repository. Each Workgroup repository has two sets of configuration properties: the Application Configuration and the Partition Configuration. The configuration properties contain much of the information that is stored in the ini files that are used on the mainframe repository. For detailed information on the client side code generation, refer to the *Deploying Applications Guide* .
This chapter describes the settings in the Application Configuration initialization (Appcfgdefault.ini) file. Refer to Partitiondefault.ini Settings for the Partition Configuration settings

Sections in the Appcfgdefault.ini

The Appcfgdefault.ini file contains the following sections. These settings are accessible through a Configuration menu available from the right-click on the Application Configuration object in the Hierarchy Window in the Construction Workbench. Refer to the *Deploying Applications Guide* for information on how to change these settings and add new sections.

[ACCMETH]	[HPSRT]	[PREPARE]
[APPCONFIG]	[ISPF]	[QUALIFIERS]
[ASM]	[JCL]	[STAGING]
[C]	[LE]	[TCPIP]
[COBOL]	[LINKAGE]	[TSODD]
[DB2]	[MVSPGM]	

[FILES]	[PLI]	

ACCMETH

This section contains the access methods settings.

Table 7-1 Appcfgdefault.ini [ACCMETH] section

Key	Possible Values / Default	Description
IDCMLIB		This setting specifies the location of IDCMPGM.
IDCMPGM	IDCAMS (default)	This setting specifies the Access Method Services commands.

APPCONFIG

This section contains the APPCONFIG settings .

Table 7-2 Appcfgdefault.ini [APPCONFIG] section

Key	Possible Values / Default	Description
ABRELEASE	<valid AppBuilder release> AB212 (default)	This setting specifies the release version of the AppBuilder.
GENID	<2 characters that are unique throughout the corporation> AA (default)	This is a two letter ID used for generating data set names.
MFCODEPG	<valid codepage> IBM037 (default)	This setting specifies the mainframe's codepage for the Application Configuration. It defaults to the value in the Workbench Options if this setting is left blank.

ASM

This section contains the Assembler parameters.

Table 7-3 Appcfgdefault.ini [ASM] section

Key	Possible Values / Default	Description
APARM1	NODECK,OBJECT (default)	This setting specifies the Assembler parameters line 1.
APARM2	LIST (default)	This setting specifies the Assembler parameters line 2.
DB2PARM	default: HOST(ASM),SOURCE,APOST,GRAPHIC	This setting specifies the Assembler DB2 parameters.
HASMMODG	SYS1.AMODGEN (default)	This setting specifies the Assembler Module library.
HASMPGM	ASMA90 (default)	This setting specifies the Assembler Compiler program.
MACLIB	SYS1.MACLIB (default)	This setting specifies the Assembler Macro library.
MODGEN	SYS1.MODGEN (default)	This setting specifies the Assembler Module library.
TRNPARM	No default value	This setting specifies the Assembler CICS transaction parameters.

_C_

This section contains the settings for C.

Table 7-4 Appcfgdefault.ini [C] section

Key	Possible Values / Default	Description
C370CPRT	default: (RECFM=VBA,LRECL=137,BLKSIZE=882)	This setting specifies the data set attributes for Compiler output.

C370HDRS	CEE.SCEEH.H (default)	This setting specifies the C370 header file.
C370LNK1	CEE.SCEERUN (default)	This setting specifies the first C370 Linkedit library.
C370LNK2	CEE.SCEEPROC (default)	This setting specifies the second C370 Linkedit library.
C370MSGS	CEE.SCEEMSGP(EDCPMSGE) (default)	This setting specifies the C370 error messages.
C370PGM	CBCDRVR (default)	This setting specifies the C370 compiler program.
C370STEP	CBC.SCBCCMP (default)	This setting specifies the C370 Step library.
C37PARM1	NOOPT,SO,OBJ,SHOWINC,TEST (default)	This setting specifies the C parameters line 1.
C37PARM2	LIST,NOMAR (default)	This setting specifies the C parameters line 2.
CSTEPI	Y (default) N	This setting specifies whether or not to use the C370 Step library.
DB2PARM	HOST(C),SOURCE,APOST (default)	This setting specifies the C DB2 parameters.
TRNPARM	No default value	This setting specifies the C CICS transaction parameters.

COBOL

This section contains the COBOL parameters.

Table 7-5 Appcfgdefault.ini [COBOL] section

Key	Possible Values / Default	Description
COB2PGM	IGYCRCTL (default)	This setting specifies the COBOL compiler program.
COB2STEP	IGY.V2R2M0.SIGYCOMP (default)	This setting specifies the COBOL step library
COB2STPI	Y (default) N	This settings specifies whether or not to Include the COBOL Step library in the Compiler step.
CPARM1	default: RES,RENT,OBJECT,LIB,NOCMPR2,'TRUNC(BIN)',NODYNAM,APOST	This setting specifies the COBOL parameters line 1. CPARM1 is needed for the COBOL component prepares.
CPARM2	default: SEQ,LIST,'FLAG(W)','DATA(31)',OPTIMIZE,AR(E)	This setting specifies the COBOL parameters line 2. CPARM2 is needed for the COBOL component prepares.
DB2PARM	default: HOST(COBOL),SOURCE,APOST,GRAPHIC	This setting specifies the COBOL DB2 parameters.
OCCPARM1	default: RES,RENT,OBJECT,LIB,NOCMPR2,'TRUNC(BIN)',NODYNAM,APOST	This setting specifies the OpenCOBOL parameters for the COBOL compile step of Rule Prepare.
OCCPARM2	default: SEQ,LIST,MAP,XREF,'FLAG(W)','DATA(31)',OPT,AR(E)	This setting specifies the OpenCOBOL parameters for the COBOL compile step of Rule Prepare.
TRNPARM	No default value	This setting specifies the COBOL CICS transaction parameters.

_DB2_

This section contains the DB2 settings and parameters.

Table 7-6 Appcfgdefault.ini [DB2] section

Key	Possible Values / Default	Description
DB2LINK	DSN610.SDSNLINK (default)	This setting specifies the DB2 Link library.
DB2LOAD	DSN610.SDSNLOAD (default)	This setting specifies the DB2 loadlib.

DB2LOADI	Y (default) N	This setting specifies whether or not to Include the DB2 LOADLIB in the Step library for the DB2 preprocessor.
DB2PRE	DSNHPC (default)	This setting specifies the DB2 preprocessor.
DFHCEE	DFHELII (default)	This setting specifies the DB2 language dependent modules for CICS. For LE environment, this must be set to DFHELII.
TEP2LIB	<data set name> example: DSN610.RUNLIB.LOAD No default value	This setting specifies the library where TEP2PGM is located. This setting is required for remote preparation.
TEP2PGM	DSNTEP2 (default)	This setting is used by bind and grant steps to run SQL.
TEP2PLAN	DSNTEP61 (default)	This setting specifies the plan under which TEP2PGM was bound.
TIADPGM	DSNTIAD (default)	This setting specifies the DB2 dynamic programs. Do NOT modify this entry.

FILES

This section contains information required to allocate data sets.

Table 7-7 Appcfgdefault.ini [FILES] section

Key	Possible Values / Default	Description
ADIRBLK	100 (default)	This setting specifies the number of Directory blocks for temporary data sets.
APRIM	45 (default)	This setting specifies the number of Primary tracks for temporary data sets.
ASEC	45 (default)	This setting specifies the number of Secondary tracks for temporary data sets.
CMP2SYBK	<numeric value> 23440 (default)	This setting specifies the COB Compile SYSLIB blksize. If 0, blksize is not included.
CMPASYBK	<numeric value> 23440 (default)	This setting specifies the ASM Compile SYSLIB blksize. If 0, blksize is not included.
CMPCSYBK	<numeric value> 23440 (default)	This setting specifies the C Compile SYSLIB blksize. If 0, blksize is not included.
CMPPSYBK	<numeric value> 23440 (default)	This setting specifies the PL/I Compile SYSLIB blksize. If 0, blksize is not included.
DBRMBLK	<numeric value> 3120 (default)	This setting specifies the standard blksize for all &&DBRMLIB output.
IEWLSYBK	<numeric value> 23200 (default)	This setting specifies the Linkedit SYSLIB blksize. If 0, blksize is not included.
PUNIT	<valid unit type> SYSDA (default)	This setting specifies the unit type for permanent data sets.
PUNITVOL	UNIT=SYSDA (default)	This setting specifies the unit and vol (if required). It is generated based on the information from the above three values.
PVOL	blank Y N NOVOL (default)	This setting specifies the volume specified for VOL=SER when allocating permanent data sets when PVOLREQ=Y.
PVOLREQ	Y N (default)	This setting specifies whether or not a vol=ser is required when allocating permanent data sets.
REC121BK	<numeric value> 23474 (default)	This setting specifies the standard BLKSIZE for all LRECL=121 data sets.
REC133BK	<numeric value> 23408 (default)	This setting specifies the standard BLKSIZE for all LRECL=133 data sets.

REC255BK	<numeric value> 5100 (default)	This setting specifies the standard BLKSIZE for all LRECL=255 data sets.
REC512FB	27648 (default)	This setting specifies the standard Blksize for all LRECL 512 data sets.
REC517BK	22748 (default)	This setting specifies the standard Blksize for all LRECL 517 data sets.
REC80BK	<numeric value> 23440 (default)	This setting specifies the standard BLKSIZE for all LRECL=80 data sets.
RLC2SYBK	<numeric value> 23200 (default)	This setting specifies the rule COB2 SYSLIB blksize. If 0, blksize is not included.
RLDBSYBK	<numeric value> 19040 (default)	This setting specifies the rule DB2 SYSLIB blksize. If 0, blksize is not included.
RLLKSYBK	<numeric value> 23200 (default)	This setting specifies the rule LINK SYSLIB blksize. If 0, blksize is not included.
SYSLINBK	<numeric value> 3120 (default)	This setting specifies the standard blksize for all SYSLIN output.
TUNIT	<valid unit type> SYSDA (default)	This setting specifies the unit type for temporary data sets.
TUNITVOL	UNIT=SYSDA (default)	This setting specifies the unit and vol (if required). It is generated based on the information from the above three values.
TVOL	<volume number or NOVOL> NOVOL (default)	This setting specifies the volume specified for vol=ser when allocating temporary data sets.
TVOLREQ	Y N (default)	This setting specifies whether or not a vol=ser is required when allocating temporary data sets.
VUNIT	<valid unit type> SYSDA (default)	This setting specifies the unit type for "virtual" data sets.
VUNITVOL	UNIT=SYSDA (default)	This setting specifies the unit and vol (if required). It is generated based on the information from the above three values.
VVOL	blank Y N NOVOL (default)	This setting specifies the volume specified for vol=ser when allocating "virtual" data sets.
VVOLREQ	Y N (default)	This setting specifies whether or not a vol=ser is required when allocating "virtual" data sets.

HPSRT

This section represent the data set suffix used during the batch runtime. It is used to create the output data set name by concatenating the standard prefix.

Table 7-8 Appcfgdefault.ini [HPSRT] section

Key	Possible Values / Default	Description
ABLOADBT	<one PDS qualifier> SLOADBT (default)	This setting specifies AppBuilder load library output for Batch.
ABLOADCI	<one PDS qualifier> SLOADCI (default)	This setting specifies AppBuilder load library output for CICS.
LOADBS	<one PDS qualifier> LOADBS (default)	This setting specifies the load library output, which is not execution environment specific.
LOADBT	<one PDS qualifier> LOADBT (default)	This setting specifies the load library output for Batch.
LOADCI	<one PDS qualifier> LOADCI (default)	This setting specifies the load library output for CICS.

NCALBS	<one PDS qualifier> NCALBS (default)	This setting specifies the NCALLIB output, which is not execution environment specific.
NCALBT	<one PDS qualifier> NCALBT (default)	This setting specifies the NCALLIB output for Batch.
NCALCI	<one PDS qualifier> NCALCI (default)	This setting specifies the NCALLIB output for CICS.

ISPF

This section contains the default ISPF settings.

Table 7-9 Appcfgdefault.ini [ISPF] section

Key	Possible Values / Default	Description
GLBCLST1	N/A	This setting specifies the AppBuilder CLIST library.
GLBCLST2	N/A	This setting specifies the AppBuilder CLIST library.
GLBCLST3	N/A	This setting specifies the AppBuilder CLIST library.
GLBCLST4	N/A	This setting specifies the AppBuilder CLIST library.
GLBCLST5	N/A	This setting specifies the AppBuilder CLIST library.
SPFLINK	ISP.SISPLOAD (default)	This setting specifies the ISPF load library.
SPFMLIB1	ISP.SISPMENU (default)	This setting specifies the SPF Message library.
SPFMLIB2	DSN610.SDSNSPFM (default)	This setting specifies the DB2 message library.
SPFMLIB3	ISF.SISFMLIB (default)	This setting specifies the SDSF Message library.
SPFMLIB4	N/A	This setting specifies the AppBuilder Message library.
SPFMLIB5	N/A	This setting specifies the AppBuilder Message library.
SPFPLIB1	N/A	This setting specifies the AppBuilder Panel library.
SPFPLIB2	ISP.SISPPENU (default)	This setting specifies the ISPF Panel library.
SPFPLIB3	DSN610.SDSNSPFP (default)	This setting specifies the DB2 Panel library.
SPFPLIB4	DSN610.SDSNPFPE (default)	This setting specifies the DB2 Panel library.
SPFPLIB5	ISF.SISFPLIB (default)	This setting specifies the SDSF Panel library.
SPFPROF	ISP.SISPTENU (default)	This setting specifies the ISPF Table library.
SPFSLIB1	ISP.SISPSENU (default)	This setting specifies the ISPF Skeleton library.
SPFSLIB2	ISP.SISPSLIB (default)	This setting specifies the ISPF Skeleton library.
SPFSLIB3	N/A	This setting specifies the ISPF Skeleton library.
SPFSLIB4	N/A	This setting specifies the ISPF Skeleton library.
SPFSLIB5	N/A	This setting specifies the ISPF Skeleton library.
SPFTLIB1	N/A	This setting specifies the AppBuilder Table library.
SPFTLIB2	ISP.SISPTENU (default)	This setting specifies the ISPF Table library.
SPFTLIB3	ISF.SISFTLIB (default)	This setting specifies the SDSF Table library.
SPFTLIB4	N/A	This setting specifies the AppBuilder Table library.
SPFTLIB5	N/A	This setting specifies the AppBuilder Table library.

USE_USER_ISPTLIB	Y N (default)	This setting specifies whether or not AppBuilder table library uses the user ISPF Table library.

JCL

This section contains the Job Control Language (JCL) related variables.

Table 7-10 Appcfgdefault.ini [JCL] section

Key	Possible Values / Default	Description
HPSPRNT	No default value	This setting specifies the local printer variable.
INTRDR_CLASS	1 (default)	This setting specifies the output class of the internal reader.
INTRDR_NAME	INTRDR (default)	This setting specifies the name of the internal reader. Use _INTRDR and VSAM will not be used to submit the job. The job number will not be known.
JOB_CHAR	M (default)	This setting specifies the default job character for batch jobs.
JOB_CLASS	1 (default) A	This setting specifies the job class for batch jobs.
JOB_CUST	Y N (default)	This setting specifies whether or not to include the client customized skel for JCL card.
JOB_CUST_IMS	Y N (default)	This setting specifies whether or not to include the client customized skel for JCL card used for IMS.
JOB_CUST_MIG	Y N (default)	This setting specifies whether or not to include the client customized skel for JCL card used for MIG.
JOB_CUST_MTH	Y N (default)	This setting specifies whether or not to include the client customized skel for JCL card used for months.
JOB_CUST_SDS	Y N (default)	This setting specifies whether or not to include the client customized skel for JCL card used for SDS.
JOB_CUST_SEC	Y N (default)	This setting specifies whether or not to include the client customized skel for JCL card used for SEC.
JOB_MSGCLASS	X (default)	This setting specifies the message class for batch jobs.
JOB_NIGHTCLA	1 (default)	This setting specifies the overnight job class for batch jobs.
JOB_PRIORITY	12 (default)	This setting specifies the job priority for batch jobs.
JOB_REGION	6M (default)	This setting specifies the job region size for batch jobs.
JOB_TEXT	JOB (default)	This setting specifies the default job description for batch jobs.
JOB_TIME	blank (default)	This setting specifies the Time parameter for batch jobs.
TWO_CHARACTER_JOB_SUFFIX	Y N (default)	This setting specifies whether or not the TWO CHARACTER JOB SUFFIX IS USED. This setting iMPLIEs that the USER IDS ARE ONLY SIX CHARACTERS.
ZSYSOUT	<valid sysout class> • (default)	This setting specifies the Sysout class for batch jobs.

_LE_

This section contains the Language Environment information settings.

Table 7-11 Appcfgdefault.ini [LE] section

Key	Possible Values / Default	Description

SCEELKED	CEE.SCEELKED (default)	This setting specifies the Language Environment link-time library.
SCEERUN	CEE.SCEERUN (default)	This setting specifies the Language Environment run-time library.
SIBMCALL		This setting specifies the support for OS PL/I PLICALLA and PLICALLB entry points.
SIBMMATH	CEE.SIBMMATH (default)	This setting specifies the PL\I math routines for PL/I Version 2.

LINKAGE

This section contains the Linkage parameters.

Table 7-12 Appcfgdefault.ini [LINKAGE] section

Key	Possible Values / Default	Description
AMODE	31 (default)	This setting specifies the Addressing Mode.
LNKPARM	LIST,XREF,DCBS,MAP (default)	This setting specifies the linkage parameters.
NCALPARM	NCAL,LIST,XREF,DCBS,MAP (default)	This setting specifies the NCAL linkage parameters.
RMODE	ANY (default)	This setting specifies the Residency Mode.

MVSPGM

This section contains the MVS program settings.

Table 7-13 Appcfgdefault.ini [MVSPGM] section

Key	Possible Values / Default	Description
IEBLIB		This setting specifies the location of IEBPGM.
IEBPGM	IEBCOPY (default)	This setting specifies the copy program.
SORTPGM	SORT (default)	This setting specifies the sort program.

PLI

This section contains the PL/I language settings.

Table 7-14 Appcfgdefault.ini [PLI] section

Key	Possible Values / Default	Description
DB2PARM	HOST(PLI),SOURCE,GRAPHIC (default)	This setting specifies the PL/I DB2 parameters.
PLIPGM	IEL0AA (default)	This setting specifies the PL/I compiler program.
PLISTEP	IEL.V1R1M1.SIELCOMP (default)	This setting specifies the PL/I Step library.
PLISTEPI	Y (default) N	This setting specifies whether or not to include the PL/I Step library in the Compiler step.
PPARM1	default: INC,OP,AG,NEST,STMT,'OPTIMIZE(TIME)'	This setting specifies the PL/I parameters line 1.
PPARM2	default: A(F)','X(F)',MAP,OF,STG,MARGINS(2,72,1)	This setting specifies the PL/I parameters line 2.
PREPARM1	default: MACRO,NOSYNTAX,MDECK,NOINSOURCE,OFFSET,GRAPHIC	This setting specifies the PL/I precompiler parameters line 1.
PREPARM2	MARGINS(2,72,1) (default)	This setting specifies the PL/I precompiler parameters line 2.
TRNPARM	default: MARGINS(2,72,1),OMARGINS(2,72,1)	This setting specifies the PL/I CICS Transaction parameters.

PREPARE

This section represent the data set suffix used during the prepares. It is used to create the output data set name by concatenating the standard prefix.

Table 7-15 Appcfgdefault.ini [PREPARE] section

Key	Possible Values / Default	Description
CMPICICS	Y N (default)	This setting enables the use of CICS statement in user components.
CPSBADSN	<one PDS qualifier> SUBASM (default)	This setting specifies the subroutine library for Assembler.
CPSBBDSN	<one PDS qualifier> SUBCOB (default)	This setting specifies the subroutine library for COBOL.
CPSBCDSN	<one PDS qualifier> SUBC (default)	This setting specifies the subroutine library for C.
CPSBPDSN	<one PDS qualifier> SUBPL1 (default)	This setting specifies the subroutine library for PL/I.
CPSRADSN	SRCASM (default)	This setting specifies the Assembler Component Source directory.
CPSRBDSN	SRCCOB (default)	This setting specifies the COBOL Component Source directory.
CPSRCDSN	SRCC (default)	This setting specifies the C Component Source directory.
CPSRPDSN	SRCPL1 (default)	This setting specifies the PL\I Component Source directory.
DCFLBDSN	<one PDS qualifier> DCLGNCOB (default)	This setting specifies the DCLGEN library for COBOL.
DCFLCDSN	<one PDS qualifier> DCLGNC (default)	This setting specifies the DCLGEN library for C.
DCFLPDSN	<one PDS qualifier> DCLGNPL1 (default)	This setting specifies the DCLGEN library for PL/I.
FLEPRC	Y N (default)	This setting specifies whether or not to generate C DCLGEN for file prepare.
FLEPRCOB	Y (default) N	This setting specifies whether or not to generate COBOL DCLGEN for file prepare.
FLEPRPL1	Y N (default)	This setting specifies whether or not to generate PL/I DCLGEN for file prepare.
GENLIST	Y N (default)	This setting specifies whether or not to keep the prepare listings in held queue.

LINKCOMP	Y N (default)	This setting specifies whether or not to generate link edit REPLACE statements for C and PL/I components during a static link. • N: generates REPLACE statements • Y: no REPLACE statements are generated This is set to N and currently not modifiable.
OCSCPDSN	<one PDS qualifier> OCSTCPY (default)	This setting specifies the Set library for OpenCOBOL in the PREPARE section.
OCSRCDSN	<one PDS qualifier> OCSRC (default)	This setting specifies the saved generated OpenCOBOL source.
OCVCPDSN	<one PDS qualifier> OCVWCPY (default)	This setting specifies the View library for OpenCOBOL in the PREPARE section.
STCDSN	<one PDS qualifier> SETC (default)	This setting specifies the Set library for C.
STCOBDSN	<one PDS qualifier> SETCOB (default)	This setting specifies the Set library for COBOL.
STERRDSN	<one PDS qualifier> SETERR (default)	This setting specifies the Set library for error files.
STPL1DSN	<one PDS qualifier> SETPL1 (default)	This setting specifies the Set library for PL/I.
STREFDSN	<one PDS qualifier> SETREF (default)	This setting specifies the Set library for reference files.
SVBNDDSN	<one PDS qualifier> SRCBNDCC (default)	This setting specifies the saved bind control cards.
SVBNTDSN	<one PDS qualifier> SRCBNDLS (default)	This setting specifies the saved bind control cards list.
SVBNXDSN	<one PDS qualifier> SRCBNDX (default)	This setting specifies the saved generated bind file in PDS.
SVBPKDSN	<one PDS qualifier> SRCBPKCC (default)	This setting specifies the saved Package Bind control cards.
SVBRRDSN	<one PDS qualifier> SRCBTRRL (default)	This setting specifies the saved batch rule relation.

SVBRSDSN	<one PDS qualifier> SRCBTRLS (default)	This setting specifies the saved batch rule source.
SVBVWDSN	<one PDS qualifier> SRCBTVW (default)	This setting specifies the saved batch viewdef.
SVCMPSRC	Y N (default)	This setting specifies whether or not to save the extracted component source. Both SVCMPSRC and SVPREP must be set to Y for component source to be saved in an external data set. The SVCMPDSN data set must also exist and be a PDS in order to get the component source saved.
SVCMVDSN	<one PDS qualifier> SRCCMPVW (default)	This setting specifies the saved extracted component view source.
SVCMVSRC	Y N (default)	This setting specifies whether or not to save the extracted component views.
SVCOBDSN	<one PDS qualifier> SRCCOBOL (default)	This setting specifies the saved generated COBOL source.
SVCPADSN	<one PDS qualifier> SRCCPASM (default)	This setting specifies the saved extracted Assembler component source.
SVCPBDSN	<one PDS qualifier> SRCCPCOB (default)	This setting specifies the saved extracted COBOL component source.
SVCPCDSN	<one PDS qualifier> SRCCPC (default)	This setting specifies the saved extracted C component source.
SVCPPDSN	<one PDS qualifier> SRCCPPL1 (default)	This setting specifies the saved extracted PL\I component source.
SVCPTDSN	<one PDS qualifier> SRCCPLS (default)	This setting specifies the saved copybook control information.
SVCRRDSN	<one PDS qualifier> SRCCIRRL (default)	This setting specifies the saved CICS rule relation.
SVCRSDSN	<one PDS qualifier> SRCCIRLS (default)	This setting specifies the saved CICS rule source.
SVCVIDSN	<one PDS qualifier> SRCCIVID (default)	This setting specifies the saved CICS vidtext.
SVCVWDSN	<one PDS qualifier> SRCCIVW (default)	This setting specifies the saved CICS viewdef.
SVFLBDSN	<one PDS qualifier> SRCDCCOB (default)	This setting specifies the saved extracted file prepare for COBOL.

SVFLCDSN	<one PDS qualifier> SRCDCC (default)	This setting specifies the saved extracted file prepare for C.
SVFLEC	Y N (default)	This setting specifies whether or not to save the generated file prepare for C.
SVFLECOB	Y N (default)	This setting specifies whether or not to save the generated file prepare for COBOL.
SVFLEPL1	Y N (default)	This setting specifies whether or not to save the generated file prepare for PL/I.
SVFLPDSN	<one PDS qualifier> SRCDCPL1 (default)	This setting specifies the saved extracted file prepare for PL/I.
SVFRMDSN	<one PDS qualifier> SRCFORM (default)	This setting specifies the saved extracted form source.
SVGPKDSN	<one PDS qualifier> SRCGPKCC (default)	This setting specifies the saved Package Grant control cards.
SVGRTDSN	<one PDS qualifier> SRCGRTCC (default)	This setting specifies the saved grant control cards.
SVLKTDSN	<one PDS qualifier> SRCLNKLS (default)	This setting specifies the saved link control cards list.
SVLNBDSN	<one PDS qualifier> SRCLNBCC (default)	This setting specifies the saved link control cards for BATCH.
SVLNCDSN	<one PDS qualifier> SRCLNCCC (default)	This setting specifies the saved link control cards for CICS.
SVLNIDSN	<one PDS qualifier> SRCLNICC (default)	This setting specifies the saved link control cards for IMS.
SVLNSDSN	<one PDS qualifier> SRCLNSCC (default)	This setting specifies the saved link control cards for non-execution environment specific.
SVLU2DSN	<one PDS qualifier> SRCLU2CT (default)	This setting specifies the saved LU2 control cards.
SVPREP	Y (default) N	This setting specifies whether or not to save the online prepare data sets. Both SVCMPSRC and SVPREP must be set to Y for component source to be saved in an external data set. The SVCMPDSN data set must also exist and be a PDS in order to get the component source saved.
SVRULDSN	<one PDS qualifier> SRCRULE (default)	This setting specifies the saved extracted rule source.
SVRULSRC	Y N (default)	This setting specifies whether or not to save the extracted Rule source.

SVSCTDSN	<one PDS qualifier> SRCSECT (default)	This setting specifies the saved extracted section source.
SVSETDSN	<one PDS qualifier> SRCSTASM (default)	This setting specifies the saved generated set source.
SVSETSRC	Y N (default)	This setting specifies whether or not to save the generated Set source.
SVSTBCPY	Y N (default)	This setting specifies whether or not to save the generated Set COBOL copybooks.
SVSTBDSN	<one PDS qualifier> SRCSTCOB (default)	This setting specifies the saved extracted set COBOL copylib.
SVSTCCPY	Y N (default)	This setting specifies whether or not to save the generated Set C copybooks.
SVSTCDSN	<one PDS qualifier> SRCSTC (default)	This setting specifies the saved extracted set C copylib.
SVSTECPY	Y N (default)	This setting specifies whether or not to save the generated Set error files.
SVSTEDSN	<one PDS qualifier> SRCSTERR (default)	This setting specifies the saved extracted set error file.
SVSTPCPY	Y N (default)	This setting specifies whether or not to save the generated Set PL/I copybooks.
SVSTPDSN	<one PDS qualifier> SRCSTPL1 (default)	This setting specifies the saved extracted set PL/I copylib.
SVSTRCPY	Y N (default)	This setting specifies whether or not to save the generated Set reference file.
SVSTRDSN	<one PDS qualifier> SRCSTREF (default)	This setting specifies the saved extracted Set reference file.
SVVAADSN	<one PDS qualifier> SRCVWASA (default)	This setting specifies the saved extracted view prepare for ASM aligned.
SVVASDSN	<one PDS qualifier> SRCVWASM (default)	This setting specifies the saved extracted view prepare for ASM.
SVVBASRC	Y N (default)	This setting specifies whether or not to save the generated aligned COBOL view.
SVVBSRC	Y (default) N	This setting specifies whether or not to save the generated COBOL view.
SVVCADSN	<one PDS qualifier> SRCVWCA (default)	This setting specifies the saved extracted view prepare for COBOL aligned.

SVVCADSN	<one PDS qualifier> SRCVWCA (default)	This setting specifies the saved extracted view prepare for C aligned.
SVVCBDSN	<one PDS qualifier> SRCVWCOB (default)	This setting specifies the saved extracted view prepare for COBOL.
SVVCDSN	<one PDS qualifier> SRCVWC (default)	This setting specifies the saved extracted view prepare for C.
SVVPADSN	<one PDS qualifier> SRCVWPLA (default)	This setting specifies the saved extracted view prepare for PL1 aligned.
SVVPASRC	Y N (default)	This setting specifies whether or not to save the generated aligned PL1 view.
SVVPLDSN	<one PDS qualifier> SRCVWPL1 (default)	This setting specifies the saved extracted view prepare for PL/I.
SVVPSRC	Y (default) N	This setting specifies whether or not to save the generated PL/I view.
VWASMDSN	<one PDS qualifier> VIEWASM (default)	This setting specifies the View library for Assembler.
VWCBADSN	<one PDS qualifier> VIEWCOBA (default)	This setting specifies the View library for COBOL aligned.
VWCOBDSN	<one PDS qualifier> VIEWCOB (default)	This setting specifies the View library for COBOL.
VWPL1DSN	<one PDS qualifier> VIEWPL1 (default)	This setting specifies the View library for PL/I.
VWPLADSN	<one PDS qualifier> VIEWPL1A (default)	This setting specifies the View library for PL/I aligned.

QUALIFIERS

This section represent the data set suffix used during the execution of the repository. It is used to create the output data set name by concatenating the standard prefix.

Table 7-16 Appcfgdefault.ini [QUALIFIERS] section

Key	Possible Values / Default	Description
BASEQUAL	<DSN qualifiers with a max length of 17> No default value	This setting specifies the prefix for AppBuilder BASE level data sets.

PERMQUAL	<DSN qualifiers with a max length of 10 (example: TEMP.PP)> No default value	This setting specifies the prefix for permanent output data sets. This is the qualifier for the staging libraries that are created when STAGING is set to Y in the [REBUILD] section of the Partitiondefault.ini file.
TEMPQUAL	<DSN qualifiers with a max length of 10 (example: TEMP.NN)> No default value	This setting specifies the prefix for temporary output data sets.
VSMQUAL	<DSN qualifiers with a max length of 20> No default value	This setting specifies the prefix for BASE VSAM data sets, which are NOT versioned.

STAGING

This section contains the information regarding the size of the staging libraries created when the STAGING setting is set to Y in the [REBUILD] section of the Partition Configuration ini file (Partitiondefault.ini). The staging library PDSE accepts the format in which the parameters are separated by spaces.

Do NOT change the record length for any of the Keys listed in this table.

Table 7-17 Appcfgdefault.ini [STAGING] section

Key	Possible Values / Default	Description
FILE_APPLDBRM	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the DBRM library.
FILE_CPSBADSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the subroutine library for Assembler.
FILE_CPSBBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the subroutine library for COBOL.
FILE_CPSBCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the subroutine library for C.
FILE_CPSBPDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the subroutine library for PL\I.
FILE_CPSRADSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies Assembler Component source.
FILE_CPSRBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies COBOL Component source.
FILE_CPSRCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies C Component source.
FILE_CPSRPDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies PL\I Component source.
FILE_DCFLBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies Dclgen library for COBOL.

FILE_DCFLCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies Dclgen library for C.
FILE_DCFLPDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies Dclgen library for PL\I.
FILE_LOADBS	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 0 23200 (default)	This setting specifies the Batch\CICS load library.
FILE_LOADBT	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 10 CYL 0 23200 (default)	This setting specifies the Batch load library.
FILE_LOADCI	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 10 CYL 0 23200 (default)	This setting specifies the CICS load library.
FILE_NCALBS	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 0 23200 (default)	This setting specifies the Batch\CICS NCAL library.
FILE_NCALBT	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 0 23200 (default)	This setting specifies the Batch NCAL library.
FILE_NCALCI	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 0 23200 (default)	This setting specifies the CICS NCAL library.
FILE_OCSCPDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the OpenCOBOL Set copybook library.
FILE_OCSRCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the OpenCOBOL source library.
FILE_OCVCPDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the OpenCOBOL View copybook library.
FILE_STCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 61 Set library for C.
FILE_STCOBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 58 Set library for COBOL.
FILE_STERRDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 59 Set library for Error Sets.
FILE_STPL1DSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 60 Set library for PL/I.
FILE_STREFDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 59 Set library for reference files.
FILE_SVBNXDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 286 5664 (default)	This setting specifies the saved generated bind file in PDS.
FILE_SVBPKDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved Package Bind control cards.
FILE_SVCPADSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted Assembler component source.

FILE_SVCPBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted COBOL component source.
FILE_SVCPCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted C component source.
FILE_SVCPPDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted PL\I component source.
FILE_SVFLBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted file prepare for COBOL.
FILE_SVFLCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted file prepare for C.
FILE_SVFLPDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted file prepare for PL/I.
FILE_SVGRTDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved grant control cards.
FILE_SVLNBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved link control cards for BATCH.
FILE_SVLNCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved link control cards for CICS.
FILE_SVLU2DSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved LU2 control cards.
FILE_SVRULDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted rule source.
FILE_SVSETDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved generated Set source.
FILE_SVSTBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted Set COBOL copylib.
FILE_SVSTCDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted Set C copylib.
FILE_SVSTPDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted set PL/I copylib.
FILE_SVSTRDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies the saved extracted Set reference file.
FILE_SVVCBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 50 saved extracted view prepare for COBOL.
FILE_SVVPLDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 52 saved extracted view prepare for PL/I.
FILE_VWCBADSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 62 COBOL View Data Set aligned.

FILE_VWCOBDSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 62 COBOL View data set.
FILE_VWPL1DSN	<number of containers to allocate, type of container (TRK or CYL), record length, Blocksize> 1 CYL 80 3120 (default)	This setting specifies 62 PL\I View data set.

TCPIP

This section contains the TCP/IP settings.

Table 7-18 Appcfgdefault.ini [TCPIP] section

Key	Possible Values / Default	Description
SEZACMTX	TCPIP.SEZACMTX (default)	This setting specifies the IBM Communications Server (IP Services component).
SEZATCP	TCPIP.SEZATCP (default)	This setting specifies the IBM Communications Server.

TSODD

This section contains values that control TSODD data set allocation.

Table 7-19 Appcfgdefault.ini [TSODD] section

Key	Possible Values / Default	Description
NTSTEPLB	Y (default) N	This setting specifies whether or not to allocate the Step library in the non-DB2 steps.
TSDB2LOD	Y N (default)	This setting specifies whether or not to include the DB2 load library in the Step library.
TSGLBCLS	Y (default) N	This setting specifies whether or not to allocate the Global clist libraries.
TSISPMLB	Y (default) N	This setting specifies whether or not to allocate the ISPF MLIB.
TSISPPLB	Y (default) N	This setting specifies whether or not to allocate the ISPF PLIB.
TSISPSLB	Y (default) N	This setting specifies whether or not to allocate the ISPF SLIB.
TSISPTLB	Y (default) N	This setting specifies whether or not to allocate the ISPF TLIB.
TSMDINI	Y N (default)	This setting specifies whether or not to include the AppBuilder mod level INI.
TSMDLLIB	Y N (default)	This setting specifies whether or not to include the AppBuilder mod level ISPF LLIB.
TSMDLOAD	Y N (default)	This setting specifies whether or not to include the AppBuilder mod level Library in the Step library.
TSMDMSG	Y N (default)	This setting specifies whether or not to include the AppBuilder mod level ISPF messages.
TSMDPNL	Y N (default)	This setting specifies whether or not to include the AppBuilder mod level ISPF panel.
TSMDSKL	Y N (default)	This setting specifies whether or not to include the AppBuilder mod level ISPF skeleton.
TSSTEPLB	Y (default) N	This setting specifies whether or not to allocate the Step library in TSODD.

Partitiondefault.ini Settings

You can use AppBuilder to generate codes on a workstation and remotely prepare objects to a mainframe using the information in the Workgroup repository rather than the Enterprise repository. Each Workgroup repository has two sets of configuration properties: the Application Configuration and the Partition Configuration. The configuration properties contain much of the information that is stored in the ini files that are used on the mainframe repository. For detailed information on the client side code generation, refer to the *Deploying Applications Guide* .
This chapter describes the settings in the following Partition Configuration initialization files.

- Sections in the Partitiondefault.ini - for Windows
- Sections in the Unixpartitiondefault.ini - for UNIX
- Sections in Aixpartitiondefault.ini - for AIX
- Sections in JavaPartitiondefault.ini - for Java

Refer to Appcfgdefault.ini Settings for the Application Configuration settings.

Sections in the Partitiondefault.ini

The Partitiondefault.ini file contains the following sections. These settings are accessible through a Configuration menu available from the right-click on the Partition object in the Hierarchy Window in the Construction Workbench. Refer to the *Deploying Applications Guide* for information on how to change these settings and add new sections.

[BATCH]	[MF_CODEGEN]	[REGION1]
[CICS]	[OPENCOBOLGEN]	[REMPREP]
[CODEGENPARAMETERS]	[PREPORT]	[TCPIP]
[DB2]	[QUALIFIERS]	[USRLIBS]
[JOB_INFO]	[REBUILD]	

[BATCH]

This section contains batch specific information.

Table 8-1 Partitiondefault.ini [BATCH] section

Key	Possible Values / Default	Description
BTCHLNK	RENT NORENT (default)	This setting specifies the Rules and the Components are linked with either the RENT or NORENT link edit parameter.
HLI	Y (default) N	This setting specifies whether or not to replace DSNHLI with HPSHLI during link edit. When it is set to N, DSNHLI is included.
RUN_CICS_TRANSLATOR	Y N (default)	This setting specifies whether or not the Components must use the CICS translator. When set to Y, it is verified that CICS.CICSRUN is set to Y.

[CICS]

This section contains specific information regarding the CICS setup for the Partition.

Table 8-2 Partitiondefault.ini [CICS] section

Key	Possible Values / Default	Description
C370TRN	DFHEDP1$ (default)	This setting specifies the CICS C370 translator.
CCOBTRN	DFHECP1$ (default)	This setting specifies the CICS COBOL translator.
CEDA	CEDA (default)	This setting specifies the transaction name for the IBM equivalent of CEDA.
CEMT	CEMT (default)	This setting specifies the transaction name for the IBM equivalent of CEMT.

CICS_ASSIGN_TRAN	DB2_RULES_ONLY (default) ALWAYS FRONTIER_ONLY	This setting determines when the Transactions are created for Remote Prepare. The frontier rule always generates a transaction if there is DB2 in the hierarchy.
CICSLCOB	<data set name> CICSTS13.CICS.SDFHCOB (default)	This setting specifies the COBOL compiler for CICS.
CICSLMAC	<data set name> CICSTS13.CICS.SDFHMAC (default)	This setting specifies the Assembler compiler for CICS.
CICSLNK	RENT NORENT (default)	This setting specifies whether the Rules and the Components are linked with the RENT or NORENT link edit parameter.
CICSLPLI	<data set name> CICSTS13.CICS.SDFHPL1 (default)	This setting specifies the PL/I compiler for CICS.
CICSREL	3 (default)	This setting specifies the release of CICS.
CICSRUNI	Y N (default)	This setting specifies whether or not to include the CICS Runtime library in the STEPLIB for the CICS translator.
CICSRUNT	<data set name> example: CICSTS13.CICS.SDFHLOAD No default value	This setting specifies the CICS runtime library. This setting is required for remote preparation.
CICSTRAN	Y (default) N	This setting specifies whether or not to use the CICS translator.
CICSVER	3 (default)	This setting specifies the version of CICS.
CLI	Y N (default)	This setting specifies whether or not to replace DSNHLI with HPSHLI during link edit. When set to N, DSNHLI is included.
CMPICICS	Y N (default)	This setting specifies to include the CICS runtime and link module for components which include CICS statements. Possible values are: Y and N (default value)
COBCLINK	<data set name> example: CEE.SCEECICS No default value	This setting specifies the CICS COBOL II link library.
CPLITRN	DFHEPP1$ (default)	This setting specifies the CICS PL/I translator.
HASMTRN	DFHEAP1$ (default)	This setting specifies the CICS ASM translator.
ONE_CICS_TRX	No default value	This setting specifies the transaction name if one transaction ID is to be used for all CICS rules.
PCIOPGM	DNAOC00 (default)	This setting specifies the AppBuilder cooperative communications program.
PSDOPGM	DNAOC00 (default)	This setting specifies the AppBuilder pseudo conversational communications program.
RDOCSDLT	Y N (default)	This setting specifies the use of the DFHCSDUP utility during update. • Y: The CSD is updated immediately. • N: The CSD is updated later.
RDODEFN	Y N (default)	This setting specifies the use of the DFHCSDUP utility during install. • Y: The RDO method is used. • N: The full LU2 method is used.

REGIONCNT	<numeric value> 1 (default)	This setting specifies the number of CICS regions to be processed. If MRO is used, then it must be specified here. There is no predefined processing for any region; they must be configured as needed for each region. Non-MRO uses one region.
TRANID_OUT_OF_RANGE	E (default) W	When the transaction is out of the Partition's range, this setting determines whether to give a (W)arning or an (E)rror message. For a warning, a new Transaction is issued.
UTBLTRAN	UTBL (default)	This setting specifies the AppBuilder transaction for updating the in-core table.
WHICH_TRANID	S T (default)	When the transaction ID on the rule is different from the transaction ID in the TRAN file on the mainframe, this setting determines whether to use the Rule's transaction ID ((S)ource) or the transaction ID that is stored in the TRAN file on MVS ((T)arget). When set to T, the rule is updated with the new transaction.

[CODEGENPARAMETERS]

Table 8-3 Partitiondefault.ini [CODEGENPARAMETERS] section

Key	Possible Values / Default	Description
CompareDatesAsString	Y N (default)	This is valid for OpenCOBOL only. This setting controls how the date comparison is generated. • Y: The date fields are compared as strings, where result is defined by the environment. • N: The date fields are converted to integer values, and the integer values are compared, which is platform independent. The string comparison is used only for the host variables (in SQL ASIS blocks, see also DATEDB2CMP) verification. In all other cases, the date values are first converted to integer values, and then compared.
DATEDB2CMP	0001-01-01 (default)	This is valid for OpenCOBOL only. This setting specifies the value to compare all host DATE variables with. See also DATEDB2DFLT setting.
DATEDB2DFLT	0001-01-01 (default)	This is valid for OpenCOBOL only. This setting specifies the value to assign to DATE host variable if variable is less than DATEDB2CMP setting.
DATEFMT	%Y-%0m-%0d (default)	This is valid for OpenCOBOL only. This format is used when the support library function is called. The user can change the delimiters only.
DATEINIT	0000-00-00 (default)	This is valid for OpenCOBOL only. This value is used to initialize fields of DATE type.
DEFAULT_CENTURY	1900 (default)	This setting specifies the value for the century in DATE function when only two digits are used for the year in the input and the format strings. This value is added to the two digit year parsed. This value is also passed to the support library functions. To change for runtime, the rule must be re-prepared.
DFLTDTFMT	%0m/%0d/%Y (default)	This is valid for ClassicCOBOL and OpenCOBOL. This format is used for DATE and CHAR functions when the second parameter is omitted.
DFLTTMFMT	%0t:%0m:%0s (default)	This is valid for ClassicCOBOL and OpenCOBOL. This format is used for TIME and CHAR functions when the second parameter is omitted.
DFLTTSFMT	%0o-%0d-%Y.%0t.%0m.%0s.%f	This setting is valid for OpenCOBOL. This format is used for timestamp and char functions when second parameter was omitted.
NLSIN	<character_sequence_1>	If both NLSIN and NLSOUT are defined, then they are used to convert all long names (from repository or from DCL ENDDCL section). If <character_sequence_1> and <character_sequence_2> have different length, the longer one will be truncated. When the long name is converted (including long names of views, rules, sets, symbols, fields, windows, etc.), all characters in the name included in NLSIN are replaced with characters from NLSOUT that have the same index. For example, the first character from NLSIN is replaced with the first character in NLSOUT. No verification is performed to ensure the resulting name is not ambiguous.

NLSOUT	<character_sequence_2>	If both NLSIN and NLSOUT are defined, then they are used to convert all long names (from repository or from DCL ENDDCL section). If <character_sequence_1> and <character_sequence_2> have different length, the longer one will be truncated. When the long name is converted (including long names of views, rules, sets, symbols, fields, windows, etc.), all characters in the name included in NLSIN are replaced with characters from NLSOUT that have the same index. For example, the first character from NLSIN is replaced with the first character in NLSOUT. No verification is performed to ensure the resulting name is not ambiguous.
SQLINITFLAG	0	This setting is supported for OpenCOBOL only. This controls whether SQL-INIT-FLAG will be reset to 0 in each rule. If setting is equal to 0 then MOVE ZERO TO SQL-INITFLAG is generated for each database rule. By default SQL-INIT-FLAG is not initialized. When COBOL SQL co-processor for DB2 is used this flag should be removed (disabled).
TIMEFMT	%0t.%0m.%0s.%0f (default)	This is valid for OpenCOBOL only. This format is used when the support library function is called. The user can change the delimiters only.
TIMEINIT	00.00.00.000 (default)	This is valid for OpenCOBOL only. This value is used to initialize fields of TIME type.
TIMESTAMPDB2CMP	default: 0001-01-01-00.00.00.000000	This is valid for OpenCOBOL only. This setting specifies the value to compare all host TIMESTAMP variables with. See also TIMESTAMPDB2DFLT setting.
TIMESTAMPDB2DFLT	default: 0001-01-01-00.00.00.000000	This is valid for OpenCOBOL only. This setting specifies the value to assign to TIMESTAMP host variable if it is less than TIMESTAMPDB2CMP setting.
TIMESTAMPFMT	%Y-%0o-%0d-%0t.%0m.%0s.%0f	This setting is valid for OpenCOBOL only. This format is used when support library function is called. Only delimiters can be changed by the end user.
TIMESTAMPINIT	default: 0000-00-00-00.00.00.000000	This is valid for OpenCOBOL only. This value is used to initialize fields of TIMESTAMP type.

[DB2]

This section contains DB2 specific information, which is Partition specific.

Table 8-4 Partitiondefault.ini [DB2] section

Key	Possible Values / Default	Description
DB2GRANT	Y (default) N	This setting specifies whether or not to execute the Grant after prepare binds.
PCKG	P (default)	This setting specifies whether or not to support the packages. Because only the PACKAGE BINDS ARE SUPPORTED FOR AppBuilder version 2.1.2, tHIS VALUE must ALWAYS BE P (package mode, project bind parameters are used).
PKLIST	W (default)	This setting specifies whether PKLIST uses an explicit list or wildcard list. Because only the package binds are supported for AppBuilder version 2.1.2, tHIS VALUE must ALWAYS BE W (wildcard list).

[JOB_INFO]

This section contains Job Control Language (JCL) job information specific to the Partition.

Table 8-5 Partitiondefault.ini [JOB_INFO] section

Key	Possible Values / Default	Description
ACCT_INFO	<valid account information> No default value	This setting specifies the account information, which is required, to be used for submission of the JCL jobs.
JOB_NAME	<must be 7 characters or smaller, 6 characters if using 2 character job suffix> No default value	This setting specifies the job name to use for submitting from the current Partition. If not populated, the userID found in the Workbench Options is used.

[MF_CODEGEN]

This section provides additional parameters for CODEGEN execution.

Table 8-6 Partitiondefault.ini [MF_CODEGEN] section

Key	Possible Values / Default	Description
CGDBCS	D (default) S	This setting specifies the DBCS indicator.
COMPCALL	Y (default) N B	This setting specifies the type of OpenCOBOL component call. • Y: pass DFHEIBLK and HPSCOMMAREA parameters. • N: pass only Input and Output view as parameters. • B: pass a dummy DFHEIBLK and DFHCOMMAREA along with an Input/Output view.
GEN_CODE_ONLY	Y N (default)	This setting determines if OpenCOBOL Rule source and Set and View copybooks are saved locally without submitting Remote Prepare.
OC_ENABLED	Y (default) N	This is a global switch, indicating whether or not executing in an OpenCOBOL environment. ONLY OpenCOBOL IS SUPPORTED FOR AppBuilder 2.1.2.
SAVE_SOURCE	Y (default) N	This setting determines if OpenCOBOL Rule source and Set and View copybooks are saved locally.
USE_OCSETVIEW_COPY	Y N (default)	This setting determines if OpenCOBOL Set and View copybooks are used for Components.

[OPENCOBOLGEN]

Table 8-7 Partitiondefault.ini [OPENCOBOLGEN] section

Key	Possible Values / Default	Description
PARM	PARAM=-VMC -fdyncall -yz	This is the list of additional parameters for the code generator. Refer to *Rules Language Reference Guide* for further explanation.

[PREPORT]

This section contains the preparation options.

Table 8-8 Partitiondefault.ini [PREPORT] section

Key	Possible Values / Default	Description
RDOLU2	Y (default) N	This setting specifies whether or not to use the AppBuilder/CICS autoinstall.

[QUALIFIERS]

This section represents the qualifiers used for the selected Partition.

Table 8-9 Partitiondefault.ini [QUALIFIERS] section

Key	Possible Values / Default	Description
GRPQUAL	<DSN qualifiers with a max length of 18>	This setting specifies the prefix for userID data sets. @E_USER.USER_ID uses the mainframe userID specified in the Construction Workbench options.
PARTQUAL	<PDS qualifier with a max length of 20> No default value	This setting specifies the prefix for Partition Permanent data sets. This must conform to the installation specification.

PARTVQUAL	<PDS qualifier with a max length of 20> No default value	This setting specifies the prefix for Partition VSAM data sets. This must conform to the installation specification.

[REBUILD]

This section contains information about the Partition's rebuild settings.

Table 8-10 Partitiondefault.ini [REBUILD] section

Key	Possible Values / Default	Description
PROD_CICSREG	<valid CICS region>	This setting specifies the Production CICS Region. Region uses the same cards as the install into the Permanent libraries. The only difference is the CICS region.
PROD_COLLECTION_ID	<valid DB2 collection ID>	This setting specifies the Production Collection Id.
PROD_QUAL	<PDS qualifier with a max length of 20>	This setting specifies the Production Library qualifier.
STAGING	Y N (default)	This setting specifies whether or not to use staging libraries.

[REGION1]

This section contains specific information regarding the processing for the Partition's region.

Table 8-11 Partitiondefault.ini [REGION1] section

Key	Possible Values / Default	Description
CSDNAME	<valid CSD data set name> No default value	This setting specifies the name of CSD output for the CICS region.
DOGRPLS	Y N (default)	This setting specifies whether or not to add the AppBuilder CICS group to the startup group list. When set to Y, GRPLOC, GRPLOCNM, and GRPSTART must also be specified. Note: Unless the group is added to the startup list, definitions will NOT be reinstalled on subsequent CICS restarts.
DONEWCPY	Y (default) N	This setting specifies whether or not to pERFORM CICS NEW COPY COMMAND IN THE SPECIFIED REGION.
DOPROG	Y (default) N	This setting specifies whether or not to pERFORM CICS DEFINE PROGram COMMAND IN THE SPECIFIED REGION.
DOTRANS	Y (default) N	This setting specifies whether or not to define the AppBuilder transaction during rule prepare.
DOUTBL	Y (default) N	This setting specifies whether or not to execute the AppBuilder UTBL transaction.
GRPLOC	A B (default)	If DOGRPLS is set to Y, this setting specifies where (in the list specified by GRPSTART) the group is added relative to the group specified in GRPLOCNM. A specifies "After", and B specifies "Before".
GRPLOCNM	No default value	This setting specifies the name of the CICS group used as reference for defining the AppBuilder CICS group.
GRPPREFX	C H (default)	This setting specifies how the group name is built. The group name is a combination of the AppBuilder project name (H) and the CICS group name (C). • H: The group name is:"(AppBuilder project name)+(CICS group name)". • C: The group name is:"(CICS group name)+(AppBuilder project name)".

GRPSTART	No default value	This setting specifies the CICS startup group list used for defining AppBuilder group to the startup.
LOGAPPL1	A06HPS1 (default)	This setting specifies the VTAM applids used for HPS LU2 processing.
LOGAPPL2	A06HPS2 (default)	This setting specifies the VTAM applids used for HPS LU2 processing.
LOGAPPL3	A06HPS3 (default)	This setting specifies the VTAM applids used for HPS LU2 processing.
PCHPSPRJ	Y (default) N	This setting specifies whether or not to use the AppBuilder project as part of the CICS group for PCTs.
PCTGROUP	HPS (default)	This setting specifies the CICS group used to define all AppBuilder PCTs.
PPHPSPRJ	Y (default) N	This setting specifies whether or not to use the AppBuilder project as part of the CICS group for PPTs.
PPTGROUP	HPS (default)	This setting specifies the CICS group used to define all AppBuilder PPTs.
REMOTSID	No default value	This setting specifies the nAME OF CICS REMOTE REGION FOR TRANSACTION DEFINITION.
TEMPGRP	H (default)	This setting specifies the prefix for temporary groups used with USERID.
USE_LU2_TEMP_GROUP	Y N (default)	This setting specifies whether or not to eliminate the LU2 install statements for temporary rebuild or Super Prepare method.

[REMPREP]

Table 8-12 Partitiondefault.ini [REMPREP] section

Key	Possible Values / Default	Description
DEBUG	N (default)	This setting specifies the debug variable.

[TCPIP]

This setting must be specified ONLY WHEN the TCPIP installation at the client site does not default these parameters for client programs, or if additional or overriding parameters need to be specified. For example, this setting must be specified when the name of the TCPIP process is modified to be something other than "TCPIP."

Table 8-13 Partitiondefault.ini [TCPIP] section

Key	Possible Values / Default	Description
SYSTCPD	<valid data set name fully qualified> No default value	This setting specifies the data set that contains the TCPIP parameters for TCPIP client programs required by the user's site.

[USRLIBS]

This section contains user information regarding non-AppBuilder that may be used in conjunction with the AppBuilder product.

Table 8-14 Partitiondefault.ini [USRLIBS] section

Key	Possible Values / Default	Description
USRCPASM	<valid data set name fully qualified> No default value	This setting specifies the user copylib for ASM.
USRCPC	<valid data set name fully qualified> No default value	This setting specifies the user copylib for C.

USRCPCOB	<valid data set name fully qualified> No default value	This setting specifies the user copylib for COBOL.
USRCPPL1	<valid data set name fully qualified> No default value	This setting specifies the user copylib for PL/I.
USRDCASM	<valid data set name fully qualified> No default value	This setting specifies the user DCLGEN for ASM.
USRDCC	<valid data set name fully qualified> No default value	This setting specifies the user DCLGEN for C.
USRDCCOB	<valid data set name fully qualified> No default value	This setting specifies the user DCLGEN for COBOL.
USRDCPL1	<valid data set name fully qualified> No default value	This setting specifies the user DCLGEN for PL/I.
USRLDBS	<valid data set name fully qualified> No default value	This setting specifies the user loadlib for any execution environment.
USRLDBT	<valid data set name fully qualified> No default value	This setting specifies the user loadlib for batch.
USRLDCI	<valid data set name fully qualified> No default value	This setting specifies the user loadlib for CICS.
USRNCLBS	<valid data set name fully qualified> No default value	This setting specifies the user NCAL for any execution environment.
USRNCLBT	<valid data set name fully qualified> No default value	This setting specifies the user NCAL batch library.
USRNCLCI	<valid data set name fully qualified> No default value	This setting specifies the user NCAL CICS library.

Sections in the Unixpartitiondefault.ini

The Unixpartitiondefault.ini file contains the following sections. These settings are used for the HP-UX platform.

[AP Unix]

This section contains default settings for the Unix platform.

Table 8-15 Unixpartitiondefault.ini [AP Unix] section

Key	Possible Values / Default	Description
COBOL_COMPILER	cob32	This setting specifies the COBOL compiler.

COBOL_COMPILER_OPTIONS	-C COPYEXT=,COB,CBL -z	This setting specifies the COBOL compiler options.	
EXECUTION_MODE	ONLINE (default value) BATCH	This setting specifies the execution mode, ONLINE or BATCH. Possible values are ONLINE (default value) and BATCH.	
PREP_DEBUG	Y N (default)	This setting specifies whether or not to use the Preparation debug.	

[CLT_CODEGEN]

This section contains client code generation settings for the Unix platform.

Table 8-16 Unixpartitiondefault.ini [CLT_CODEGEN] section

Key	Possible Values / Default	Description
COMPCALL	Y (default) N B	This setting specifies the type of OpenCOBOL Component call. • Y: pass DFHEIBLK and HPSCOMMAREA parameters. • N: pass only INPUT and OUTPUT view as parameters. • B: pass a dummy DFHEIBLK and DFHCOMMAREA along with an Input/Output view.
GEN_CODE_ONLY	Y N (default)	This setting determines if OpenCOBOL Rule source and Set and View copybooks are saved locally without submitting Remote Prepare.
SAVE_SOURCE	Y N (default)	This setting specifies whether or not to save the source and copybooks on the client.
USE_OCSETVIEW_COPY	Y (default) N	This setting specifies whether or not to generate OpenCOBOL or runtime copybooks.

[MacroDomains]

Table 8-17 Unixpartitiondefault.ini [MacroDomains] section

Key	Possible Values / Default	Description
LANGUAGE	Java C Cobol OpenCobol	This setting specifies all the possible values for macro LANGUAGE.
ENVIRONMENT	Server HTML GUI	This setting specifies all the possible values for macro ENVIRONMENT.

[OPENCOBOLGEN]

This section contains OpenCOBOL generation settings for the Unix platform.

Table 8-18 Unixpartitiondefault.ini [OPENCOBOLGEN] section

Key	Possible Values / Default	Description
MACRO	ENVIRONMENT=Server	This setting defines a macro that is environment specific.
MACRO	LANGUAGE=OpenCobol	This setting defines a macro that is language specific.
PARM	PARAM= -fc5set -fdyncall -fUNIX -frtdti	This is the list of additional parameters for code generator. Requires at a minimum PARAM=, regardless if any other values are set. Refer to *Rules Language Reference Guide* for additional information.

Sections in Aixpartitiondefault.ini

The aixpartitiondefault.ini file contains the following sections.

[AP Unix]	[MacroDomains]
[CLT_CODEGEN]	[CODEGENPARAMETERS]
[OPENCOBOLGEN]	

[AP Unix]

This section contains default settings for the Unix platform.

Table 8-19 Aixpartitiondefault.ini [AP Unix] section

Key	Possible Values / Default	Description
COBOL_COMPILER	cob2_r	This setting specifies the COBOL compiler.
COBOL_COMPILER_OPTIONS	-qAPOST -qLIST -e _iwz_cobol_main	This setting specifies the COBOL compiler options.
LINK_OPTIONS	-L/usr/lib/dce -ldcelibc_r -ldcepthreads	This setting specifies the link options.
PREP_DEBUG	Y N (default)	This setting specifies whether or not to use the Preparation debug.
EXECUTION_MODE	CICS (default) BATCH	This setting specifies the execution mode.

[CLT_CODEGEN]

This section contains default settings for is for Client Codegen.

Table 8-20 Aixpartitiondefault.ini [CLT_CODEGEN] section

Key	Possible Values / Default	Description
GEN_CODE_ONLY	Y (default) N	Determines if OpenCOBOL Rule Source and Set and View Copybooks are saved locally without submitting Remote prepare.
COMPCALL	Y (default) N B	Determines the type of OpenCOBOL component call.
SAVE_SOURCE	Y (default) N	Save the source and copybooks on the client.
USE_OCSETVIEW_COPY	Y N (default)	Generate OpenCOBOL or runtime Copybooks.

[OPENCOBOLGEN]

This section is used for OpenCOBOL generation settings. This is the list of additional parameters for code generator. Refer to *Rules Language Reference Guide* for a description. Requires at a minimum PARAM=, regardless if any other values are set.

Table 8-21 Aixpartitiondefault.ini [OPENCOBOLGEN] section

Key	Possible Values / Default	Description
PARM	PARAM= -fc5set -fdyncall -fAIX -frtdti	
R2C_STANDARD_TABS	{install_dir}\ad\cfg\cg\aix\ocob.tab	
MACRO	ENVIRONMENT=Server	
MACRO	LANGUAGE=OpenCobol	

[MacroDomains]

The settings in the [MacroDomains] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility and are language and platform independent.

Table 8-22 Aixpartiondefault.ini [MacroDomains] section

Key	Possible Values / Default	Description
ENVIRONMENT	Server,HTML,GUI	This setting specifies all the possible values for macro ENVIRONMENT.
LANGUAGE	Java,C,Cobol,OpenCobol	This setting specifies all the possible values for macro LANGUAGE.

[CODEGENPARAMETERS]

The settings in the [CodeGenParameters] section are used for the Construction Workbench and Preparation & Test Server. These settings are used by the Code Generation facility, and control how the code is generated. The settings in this section are language and platform independent.

Table 8-23 Aixpartiondefault.ini [CODEGENPARAMETERS] section

Key	Possible Values / Default	Description
DFLTDTFMT	%0m/%0d/%Y	It is valid for Classic COBOL and OpenCOBOL. This format is used for date and char functions when second parameter was omitted.
DFLTTMFMT	%0t:%0m:%0s	It is valid for Classic COBOL and OpenCOBOL. This format is used for time and char functions when second parameter was omitted.
DATEFMT	%Y-%0m-%0d	It is valid for OpenCOBOL only. This format is used when support library function is called. Only delimiters can be changed by the end user.
DATEINIT	0000-00-00	It is valid for OpenCOBOL only. This value is used to initialize fields of DATE type.
TIMEFMT	%0t.%0m.%0s.%0f	It is valid for OpenCOBOL only. This format is used when support library function is called. Only delimiters can be changed by the end user.
TIMEINIT	00.00.00.000	It is valid for OpenCOBOL only. This value is used to initialize fields of TIME type.
TIMESTAMPINIT	0000-00-00-00.00.00.000000	It is valid for OpenCOBOL only. This value is used to initialize fields of TIMESTAMP type.
DEFAULT_CENTURY	1900	Sets the value for the century in DATE function when only two digits are used for the year in the input and format strings. This value would be added to the two digit year parsed. Default value is 1900. This value is also passed to the support library functions. To change it for run time the rule must be re-prepared.
DATEDB2CMP	0001-01-01	It is valid for OpenCOBOL only. The value to compare all host DATE variables with. See DATEDB2DFLT for more details.
DATEDB2DFLT	0001-01-01	It is valid for OpenCOBOL only. The value to assign to DATE host variable if variable is less than DATEDB2CMP.
TIMESTAMPDB2CMP	0001-01-01-00.00.00.000000	It is valid for OpenCOBOL only. The value to compare all host TIMESTAMP variables with. See TIMESTAMPDB2DFLT for more details.
TIMESTAMPDB2DFLT	0001-01-01-00.00.00.000000	It is valid for OpenCOBOL only. The value to assign to TIMESTAMP host variable if it is less than TIMESTAMPDB2CMP.
CompareDatesAsString	Y N (default)	It is valid for OpenCOBOL only. This setting controls how date and time fields comparison will be generated. This flag affects the comparison of DATE and TIME fields everywhere in the rules code including the cases when a host variable of DATE type is used within SQL ASIS blocks and its verification is enabled (see codegen key -yz). If set to Y then date/time fields will be compared as strings, where result is defined by the environment. When set to N date/time fields will be converted to integer values and then integer values will be compared which is platform independent.

NLSIN	<character_sequence_1>	If both values NLSIN and NLSOUT are defined then they are used to convert all long names (from repository or from DCL ENDDCL section). If character_sequence_1 and character_sequence_2 have different length then the longer one will be truncated. When long name is converted (including long names of views, rules, sets, symbols, fields, windows, etc.) all characters in the name included in NLSIN will be replaced with characters from NLSOUT which have the same index, i.e. first character from NLSIN will be replaced with the first character in NLSOUT and so on. No verification is done that the resulting name is not ambiguous.
NLSOUT	<character_sequence_2>	
SQLINITFLAG	0	It is valid for OpenCOBOL only. SQLINITFLAG setting controls whether SQL-INIT-FLAG will be reset to 0 in each rule. If setting is equal to 0 then MOVE ZERO TO SQL-INIT-FLAG. The following will be generated for each database rule. By default SQL-INIT-FLAG is not initialized.

Sections in JavaPartitiondefault.ini

JavaPartitiondefault.ini contains the following sections:

- [JavaPackage]
- [CodegenParameters]

[JavaPackage]

The settings in [JavaPackage] section are described in JavaPartitiondefault.ini [JavaPackage] section.

Tble 8-24 JavaPartitiondefault.ini [JavaPackage] section

Key	Possible Values / Default	Description
INCLUDEINPKGCMD		Use it to specify a command to run before the package is created. For example: `INCLUDEINPKGCMD=copy "%HPSDIR%\java\rt\appbuilder.ini"` When the above command is executed the current appbuilder.ini file will be copied to the directory which is the source for the Java archive.

[CodegenParameters]

The settings in [CodegenParameters] are described in JavaPartitiondefault.ini [CodegenParameters] section.

Table 8-25 JavaPartitiondefault.ini [CodegenParameters] section

Key	Possible Values / Default	Description
ASSERT_VIEW_IDENTITY	YES (default) NO	This setting specifies if CodeGen generates assertIdentity calls to ensure the use of the proper version of the views in the rule. The parameter also specifies whether to generate getHash() method in view classes. The default value is YES. The setting is only applicable for Java generation.
EXPAND_RULE_SIGNATURE	YES NO (default)	This setting expands the method's signature.
GENERATE_IO_VIEW_TRACE	YES NO (default)	This setting is set to generate input/output view debug TRACE statements. It can have the values YES or NO. When set to YES, the input view trace is generated at the beginning of the rule and the output view trace is generated at the end of the rule.The input view is written to the trace on entry to the rule, and the output view, on exit of the rule.
GENERATE_RULE_CALLS_TRACE	YES NO (default)	When GENERATE_RULE_CALLS_TRACE is set to YES debug level, TRACE statements are generated at the beginning and end of each rule. When it is set to NO, debug level TRACE statements are not generated. Debug level TRACE statements are generated conditionally, so these statements can be disabled at runtime if the debug option is turned off.

GENERATE_STATELESS_RULE	YES NO (default)	This setting makes the target stateless.
GENERATE_VIEW_FIELD_ACCESSORS	YES NO (default)	Public field accessors will be generated for views. Possible Values are YES and NO (default value).
INLINE_VIEW_COPY	YES NO (default)	If is set to YES, then the sequence of Java statements corresponding to rules MAP statement with view arguments is generated in the rule class as is (inline). If is set to NO, then this sequence is enclosed to a private rule class method and MAP is generated as call to the method (see also INLINE_VIEW_COPY_FIELDS_LIMIT).
JAVA_PERSISTENT_CURSOR	YES NO (default)	This setting controls the SQL cursor persistence. If it is set to YES, then an SQL cursor will be persistent. Persistence means that a cursor created by a particular rule must be retained after the end of the rule invocation and made available to subsequent invocations of that rule within the same scope until explicitly closed, or in the case of executing in a server request scope, the request terminates. If it is set to YES, it is equivalent to setting command line flag SQLPERSIST. Possible values are YES and NO (default value).
LAZY_INSTANTIATION_ENABLED	YES NO (default)	This parameter affects Java generation only. If this parameter is set to YES, all local variables and views (excluding redefined views and input/output views) are instantiated when they are explicitly accessed for the first time. Refer to the Code Generation and Parameters Settings section of the *Rules Language Reference Guide* .

Bphx.sdf.ini Settings

Bphx.Sdf.ini Settings

The Bphx.Sdf.ini file contains initialization information on the operation and configuration of the C#.NET applications. This .ini file as well as the Font.ini file (see Font.ini Settings) must be properly set in order to execute AppBuilder C#.NET applications.
The Bphx.Sdf.ini is located in the DotNET\sys\bin folder under the AppBuilder install directory and is copied to every partition folder during the preparation process, when any rule that belongs tot the partition is prepared for the first time. You can edit only the copy of the Bphx.Sdf.ini file located under the partition folder. Bphx.Sdf.ini file must be located in the same folder as the root rule when the application starts. It also can be renamed to match the rule's DLL name, but with the .ini extension.
The settings in the Bphx.Sdf.ini affect the C#.NET Client applications.

Sections in the Bphx.Sdf.ini file

The Bphx.Sdf.ini file contains the following sections.

[DataBase]	[Routes]
[DotNetEncodingMappings]	[SERVER.NetEssentialEntry]
[General]	[SERVER.WebServiceEntry]
[Gui]	[SERVER.WCFServiceEntry]
[Module]	[Trace]
[Client]	[Validation]
[Server]	[UserExits]

{install_dir} in the Possible Value/Default column indicates the AppBuilder installation directory.

[General]

The settings in [General] section are described in Bphx.Sdf.ini [General] section.

Table 9-1 Bphx.Sdf.ini [General] section

Key	Possible Values / Default	Description
AbortOnCommError	True (default value) False	This setting allows y (terminate) AppBuild occurs. In the thin-cli COMM_ERROR_RU any other thin-client AbortOnCommError, UseCommErrorRule two settings are mut
AuthenticationType	Exit (default value) Dialog None	This setting specifies used for the situatior • when Query true • when query/ • when a rem When is set entered inte When is set the .Net Exit When is set is used on a Possible val Dialog, and
CharCoordinateFont	CCS_FONT	This setting specifies fontmetrics for char is CCS_FONT. A section with the en default) must be defi
CheckForIllegalCrossThreadCalls	true (default value) false	This setting specifies CHAR(date1, format the format is '%0y' o
CommErrorRuleClass		User-defined HPS_C property takes a plac True, this proprty is error occurs. In this Otherwise (if the pro HPS_COMM_ERRO to standard logic of implement Bphx.Sdf. AssemblyQualifiedN name>, <assembly <token>)
DefaultCentury	1900	This setting specifies CHAR(date1, format the format is '%0y' o
DefaultDateFormat	%m/%d/%Y	This setting defines t used as display form date values in place setting. If no default specific setting is use
DefaultTimeFormat	%0h:%0m:%0s	This setting defines t used as display form time data in place of If no default time forr setting is used.
ExceptionOnUnsupportedComponents	True (default value) False	This setting specifies when unsupported c

FloatingPointStandard	IEEE754	Specifies floating-poi or HEXADECIMAL. MAX F MAX DIGITS IN FLC IEEE754 3 E38 HEXADECIMAL E75 MAX D MAX DIGITS IN FLC IEEE754 1 E308 17 HEXADECIMAL E75 18
FontsIniFileName	{install_dir}\font.ini	This setting specifies The Font.ini file must FONT_INI_URL key directories included i This key will initially l <HPSDIR>.
LoadIniFileForRootRule	True (default value) False	Set this flag to true if configuration. Config long name of root rul
QueryAuthenticationOnStartup	True False (default value)	This setting specifies on startup. Default value is false
RemoteRuleTimeout	100	This setting specifies timeout.
UseCommErrorRule	True False (default value)	In the thin-client envi the HPS_COMM_EF rule like any other thi UseCommErrorRule. AbortOnCommError two settings are mutu
UserExitClass	UserExitExample	This setting specifies user exits. It should i Bphx.Sdf.Common.I
XmlDeserializerClass		This setting specifies in Web Services. It s Bphx.Sdf.Common.N class.
XmlSerializerClass	on Client: Bphx.Sdf.Common.Net.WebService.XmlSerializer, Bphx.Sdf.Sync (default value) Bphx.Sdf.Common.Net.WebService.XmlSerializerAllNodes, Bphx.Sdf.Sync Bphx.Sdf.Common.Net.WebService.XmlSerializerEmptyLong, Bphx.Sdf.Sync Bphx.Sdf.Common.Net.WebService.XmlSerializerEmptyShort, Bphx.Sdf.Sync or custom on Server or Gateway: Bphx.Sdf.Common.Net.WebService.XmlSerializer, Bphx.Sdf (default value) Bphx.Sdf.Common.Net.WebService.XmlSerializerAllNodes, Bphx.Sdf Bphx.Sdf.Common.Net.WebService.XmlSerializerEmptyLong, Bphx.Sdf Bphx.Sdf.Common.Net.WebService.XmlSerializerEmptyShort, Bphx.Sdf or custom	This setting specifies Web Services. It sho Bphx.Sdf.Common.N class. You can completely by specifying your ov shipped classes: XmlSerializer (or not that are not IsClear, XmlSerializerAllNode elements that are IsC XmlSerializerEmptyL elements that are IsC are created (<abc>< XmlSerializerEmptyS elements that are IsC are created (<abc/>)

[Module]

The settings in [Module] section are described in Bphx.Sdf.ini [Module] section.

Table 9-2 Bphx.Sdf.ini [Module] section

Key	Possible Values / Default	Description	
GuiManagerClass	Bphx.Sdf.Client.Common.GuiManager, Bphx.Sdf.Client	AssemblyQualifiedName has the format: (<class name>, <assembly name>, <version>, <culture>, <token>). For example: Bphx.Sdf.Common.GuiManager, Bphx.Sdf AssemblyQualifiedName of the class which implements IGuiManager interface.	
WpfGuiManagerClass	Bphx.Sdf.Client.Common.SdfGuiManager, Bphx.Sdf.Client.Wpf	AssemblyQualifiedName of the class which implements IGuiManager interface for WPF.	

[Client]

The settings in [Client] section are described in Bphx.Sdf.ini [Client] section.

Table 9-3 Bphx.Sdf.ini [Client] section

Key	Possible Values / Default	Description	
GetModulesFromCurrentAssembly	true false (default value)	This setting notifies the standard implementation of the module loader to use Module class name for module assembly name or use current assembly.	
ModuleFactoryClass	Bphx.Sdf.Common.ModuleFactory, Bphx.Sdf	This setting specifies the AssemblyQualifiedName of the class which implements IModuleFactory interface.	
ModuleLoaderClass	Bphx.Sdf.Common.ModuleLoader, Bphx.Sdf	This setting specifies the AssemblyQualifiedName of the class which implements IModuleLoader interface.	

[Server]

The settings in [Server] section are described in Bphx.Sdf.ini [Server] section.

Table 9-4 Bphx.Sdf.ini [Server] section

Key	Possible Values / Default	Description	
GetModulesFromCurrentAssembly	true false (default value)	This setting notifies the standard implementation of the module loader to use Module class name for module assembly name or use current assembly.	
ModuleFactoryClass	Bphx.Sdf.Common.ModuleFactory, Bphx.Sdf	This setting specifies the AssemblyQualifiedName of the class which implements IModuleFactory interface.	
ModuleLoaderClass	Bphx.Sdf.Common.ModuleLoader, Bphx.Sdf	This setting specifies the AssemblyQualifiedName of the class which implements IModuleLoader interface.	

[UserExits]

The settings in [UserExits] section are described in Bphx.Sdf.ini [UserExits] section.

Table 9-5 Bphx.Sdf.ini [UserExits] section

Key	Possible Values / Default	Description	
AuthenticationExitClass		Specify the .Net class containing the custom authentication exit. Should implement Bphx.Sdf.Common.IClientExit interface. Example: AuthenticationExitClass=Bphx.Sdf.Samples.ClientExit, Bphx.Sdf.Samples	
EncryptionExitClass		Specify the .Net class containing the custom encryption exit for NetE. Encryption exit may encode the password and/or the entire message. Should implement Bphx.Sdf.Common.IClientExit interface. Example: EncryptionExitClass=Bphx.Sdf.Samples.ClientExit, Bphx.Sdf.Samples	

CommErrorExitClass		Specify the .Net class containing the custom exit for processing communication errors. Should implement Bphx.Sdf.Common.IClientExit interface. Example: CommErrorExitClass=Bphx.Sdf.Samples.ClientExit, Bphx.Sdf.Samples	
RoutingExitClass		Specify the .Net class containing the custom exit for processing routing. Should implement Bphx.Sdf.Common.IRouteExit interface. Example: RoutingExitClass=Bphx.Sdf.Samples.CustomRouteExit, Bphx.Sdf.Samples	
RemoteExecuteExitClass		Specify the .Net class containing the custom exit for processing remote rule execution. Should implement Bphx.Sdf.Common.IRemoteExecuteExit interface. Example: RoutingExitClass=Bphx.Sdf.Server.Samples.CustomRemoteExecuteExit, Bphx.Sdf.Server.Samples	

[Validation]

The settings in [Validation] section are described in Bphx.Sdf.ini [Validation] section.

Table 9-6 Bphx.Sdf.ini [Validation] section

Key	Possible Values / Default	Description	
CustomStringValidator		Specify the .Net class containing the custom field validation. Should implement Bphx.Sdf.Client.Common.ICustomStringValidator interface. Example: CustomStringValidator=Bphx.Sdf.Samples.CustomValidator, Bphx.Sdf.Samples	

[Gui]

The settings in [Gui] section are described in Bphx.Sdf.ini [Gui] section.

Table 9-7 Bphx.Sdf.ini [Gui] section

Key	Possible Values / Default	Description
BlockHandlerThread	True False (default value)	This key indicates whether Gui event handlers started in a separate thread are to be stopped after call to GuiManager.DisplayControl() waiting until OnModuleEnd() event handler is called. It is needed for cases when a GuiManager class does not call a Show(), ShowDialog(), ShowModal() or similar functions that have their own message loop.
CheckForIllegalCrossThreadCalls	True (default value) False	Specify whether cross thread calls are illegal for Windows Forms controls
DefaultButtonBackColor		This setting specifies the default button color to be used for buttons back color.
DefaultResourcePath	../Resource	This setting specifies the path for resource. It will be ignored if LoadResourcesFromAssembly is set to true.
EditFieldEnabledMode	Default	Specifies edit field Enabled property behaviour when property is set to false: Default - default Windows Forms behaviour, text is greyed out, this is default value Custom - AppBuilder specific behaviour, text color is not changed, field is ReadOnly, focus cannot be set using TAB key, but the field can be selected.
EnforceFocusWhenInvalid	True (default value) False	In case of control contents validated and considered invalid, this option sets focus to this control to help a user correct the data.
LoadResourcesFromAssembly	True (default value) False	This setting loads resources from the same .NET assembly as code, rather than from separate files.
PanelAutoScale	True False (default value)	When this flag is set to TRUE then AutoScaleMode=AutoScaleMode.Font and AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F)

ShowZeroOnNull	True False (default value)	Specifies whether to display zeros when value is null or nothing. Default is false(nothing)
StartSeparateThreadForUseRuleHandler	True False (default value)	This key indicates whether Gui event handlers containing "USE RULE" statement are to be started in the same (main GUI) thread or in a separate thread. It is needed for cases USE RULE is not guaranteed to return immediately (e.g the rule is executed on a remote server)
UseComboboxSelEvent	True False (default value)	Works same as COMBOBOX_SEL_EVENT flag in appbuilder.ini for Java: When set to TRUE, a combo box raises a Converse event with EventParam propery set to HPS_SELECT when a selection is made
UserControlMode	True False (default value)	This key indicates whether AppBuilder windows are generated as user controls to be embedded into other forms (either Windows Forms or WPF). Should UserControlMode be true, the end user must provide its own GuiManager that handles the placement of the UserControls See the GuiManagerClass key in the Module section.

[Trace]

The settings in [Trace] section are described in Bphx.Sdf.ini [Trace] section.

Table 9-8 Bphx.Sdf.ini [Trace] section

Key	Possible Values / Default	Description
AppLevel	All Info (default value) Debug Warning Error Fatal No	This setting specifies the application logger.
CustomLoggerConfigFile	Bphx.Sdf.config	This setting specifies the log4net configure.
ErrLevel	All Info (default value) Debug Warning Error Fatal No	This setting specifies the error log level.
LogFilters	All (default value) Info Debug Warning Error Fatal No	This setting specifies the log filters separated by ' | '. Other values are actually members of Bphx.Sdf.Common.Logging.LogFilters emun. This setting is set to All by default and is used for debugging purposes only.
Logger	Default (default value) Custom	"Default" is the simple logger implemented in Dphx.Sdf runtime.
SysLevel	All Info (default value) Debug Warning Error Fatal No	This setting specifies the system logger (runtime trace here).

[DotNetEncodingMappings]

The settings in the [DotNetEncodingMappings] section are described in Bphx.Sdf.ini [DotNetEncodingMappings] section.

Table 9-9 Bphx.Sdf.ini [DotNetEncodingMappings] section

Key	Possible Values / Default	Description
CP037	IBM037	

[SERVER.WebServiceEntry]

The settings in [SERVER.WebServiceEntry] section are described in Bphx.Sdf.ini [SERVER.WebServiceEntry] section.

Table 9-10 Bphx.Sdf.ini [SERVER.WebServiceEntry] section

Key	Possible Values / Default	Description
HOST_URL	http: //hostname:port/ServletContext/ServletName	
TYPE	WEBSERVICE	
CALL_NATIVE_SERVICE	FALSE	
NATIVE_SERVICE_NAME	IService1	
NATIVE_SERVICE_NAMESPACE	http://tempuri.org/	

[SERVER.WCFServiceEntry]

The settings in [SERVER.WCFServiceEntry] section are described in Bphx.Sdf.ini [SERVER.WCFServiceEntry] section.

Table 9-11 Bphx.Sdf.ini [SERVER.WCFServiceEntry] section

Key	Possible Values / Default	Description
TYPE	WCFSERVICE	
ENDPOINT	WCFServiceEndpoint	

[SERVER.NetEssentialEntry]

The settings in [SERVER.NetEssentialEntry] section are described in Bphx.Sdf.ini [SERVER.WebServiceEntry] section.

Table 9-12 Bphx.Sdf.ini [SERVER.WebServiceEntry] section

Key	Possible Values / Default	Description
CODEPAGE		
HOST_NAME		
PORT		
PROTOCOL		
SERVERID		
TYPE	NETE	

[Routes]

The settings in [Routes] section are described in Bphx.Sdf.ini [Routes] section.

Table 9-13 Bphx.Sdf.ini [Routes] section

Key	Possible Values / Default	Description

$ANY	WebServiceEntry	This setting specifies the default server entry for all remote rules.

[DataBase]

The following parameters described in the following table are needed to tune up the SQL connection mechanisms:

Table 9-14 Bphx.Sdf.ini [DataBase] section

Key	Possible Values/Default	Description
DBMS	<database management system name>	This setting allows you to choose from the DBMS family: MSSQL, Oracle, DB/2, Sybase, Generic, etc.
Server	<server name>	The name of the DB server.
DataBaseName	<database name>	The name of the database.
Login	<login name>	The DB login account name.
Password	<password>	The DB account password.
DBParams	Trusted_Connection=True	This setting specifies the additional DB connection parameters (e.g. Trusted_Connection=True). For DBMS=Generic all parameters are passed via connection string using DBParams value. Server, Login, etc. parameters are ignored. At least ODBC data source name must be specified, e.g. DBParams=DSN=TESTDATA;
ConnectionAssembly	<connection assembly name>	Connection assembly name where connection class is defined, unless the default assembly for corresponding DBMS is used. Connection assembly name should be fully qualified, e.g. (for DBMS=Generic) System.Data, Version=2.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089
ConnectionClass	<connection class>	Connection ADO.NET class to instantiate, unless the default class for corresponding DBMS is used. Class name should be represented as reference name, e.g. (for DBMS=Generic) System.DataOdbc.OdbcConnection

Services.ini Settings

The Services.ini file is installed in JAVA\RT\Properties under the AppBuilder install directory. The Services.ini file is dynamically generated by the preparation of AppBuilder OO Web services. You have to copy the generated Services.ini file in ServiceClient environment for invoking the Services.

Sections in the Services.ini file

The Services.ini file contains only the [SERVICES] section.

{install_dir} in the Possible Value/Default column indicates the AppBuilder installation directory.

[SERVICES]

The settings in [NC] section are described in Services.ini [NC] section.

Table 11-1 Services.ini [NC] section

Key	Possible Values / Default	Description
MYSERVICE.serviceType	WS (default) EJB RMI	This setting specifies the type of Service that is prepared. Available options are: WS, EJB, RMI.

MYSERVICE.packageName	servicepkg	This setting specifies the package name for the prepared service.
MYSERVICE.interfaceName	MYSERVICE	This setting specifies the Interface name for the service.
MYSERVICE.WSDLURL	http: //localhost:8080/myservice_ws/myservice?wsdl	This setting specifies the location of the WebService WSDL file url. Note: This settings must be updated by the user.
MYSERVICE.nameSpaceUrl	http: //servicepkg/	This setting specifies the namespace url for the Service.
MYSERVICE.serviceName	MYSERVICEService	This setting specifies the name of the service.
MYSERVICE.portName	MYSERVICEPort	This setting specifies the port name for the service.