

June 12, 2012

Design⁺⁺ 9.0 Release Notes

Design Power is pleased to announce its first independent Design⁺⁺ release. Design⁺⁺ 9.0 is backward compatible with the last Bentley Design⁺⁺ V8i (SELECTseries 2) release. No changes are needed in your application source code – just recompile it.

Highlights of the new release include the following:

Latest IDE Version

Integrated Development Environment (IDE) has gone through another major improvement cycle - numerous errors and stability issues have been fixed. Extra effort was put in to optimization resulting in some notable performance improvements.

New Licensing Mechanism with Much Improved Network Licensing

Independent Server Setup - Any workstation can be configured as a license server without installing Design⁺⁺ to it.

True Network Licensing - Design⁺⁺ finds its license anywhere on the company network automatically. And, there is no need to share server directories.

Redundant License Servers - If one license server goes offline, Design⁺⁺ can automatically find and switch to another server, on the fly.

Network License Control - License server can specify which domains, workstations, or users are included or excluded from its Design⁺⁺ license. Likewise, client workstation can specify which license servers Design⁺⁺ can obtain its license from.

Support of Latest MicroStation and AutoCAD Versions

Design⁺⁺ 9.0 integrates with the following CAD versions.

MicroStation: V7 (aka J), V8 XM Edition, V8i, V8i (SELECTseries 1, 2, & 3)

AutoCAD (32-bit): 2004-2013

AutoCAD (64-bit): 2009-2013

For additional enhancements and upgrades in Design⁺⁺ 9.0, read the rest of these Release Notes.

We are pleased to deliver the full power of Design⁺⁺ to your project. We thank you for using our products and services and appreciate the opportunity to serve you.

Yours Sincerely,

Design⁺⁺ Product Team
Design Power, Inc.

Design++ Core

- The new Lisp binary file extensions are 32w82fasl (32-bit Lisp compiler) and 64w82fasl (64-bit Lisp compiler).
- Logical host d++temp is fixed to default to TMP in case DPPTMP is undefined.
- Saving library rules to a file is modified to make sure that the rules\bin directory exists.
- Default GEO_LEVEL value is changed to 0 (current Level/Layer) from 1.
- Attribute value redetermination is fixed to allow the attribute value to be removed in case the redetermination requests another attribute's value from the user, who decided to cancel the request. Removing the attribute value being redetermined is important as it assures the attribute value remains consistent with its design rule.
- Change propagation is fixed to allow attribute value removal propagation to take place outside the context of redetermination. While in redetermination, attribute value removals are handled by delaying the value redeterminations.
- Updating IDE by Design++ is optimized significantly by delaying all IDE update messages until the change propagation is completed and then filtering out duplicate messages. You can now turn off the delayed IDE updating by evaluating (setf s::*delayed-ide-updating-on-p* nil)
- Instance creation is fixed to make sure that every instance has the *index* attribute. An instance did not have the *index* attribute if it was created from a class that was not a descendant of either *assemblies* or *parts*.

User Interface Server (UIS)

- All dialogs can now be localized.
- Modified to use the same error reporting dialog as IDE.
- Added an option to modify list dialog's OK button title.
- Added an option to modify button titles in multiple menu dialog.
- Added an option to set titles for menu select and cancel buttons.
- All dialogs now check if previously saved dialog location is viewable with the current monitor setting before reusing the location.
- Fixed pose dialog prompt concatenate/update when no previous prompt was set.
- Pose dialog used to fail if a call to dpp-prompt-ongoing-dismiss was quickly followed by a new call to dpp-prompt-ongoing-pose. This issue is now fixed.

- ‘Pose Logo’ dialog is fixed so that it won't throw an exception when there is a change of opacity while the dialog is invisible. Interestingly, this exception happened only on older Windows XP machines but not on Windows Vista, Windows 7 or Windows XP with a recent SP.

Integrated Development Environment (IDE)

Integrated Development Environment (IDE) has gone through another major improvement cycle - numerous errors and stability issues have been fixed. Extra effort was put in to optimization resulting in some notable performance improvements.

- IDE settings are now stored in <user>\AppData\Local\Design Parametrics\Design⁺⁺
- Refresh command is fixed to update also inheritance link information for links that are changed outside of IDE. If a component associated with the link change is already selected in Component Editor, then the update is automatic. Otherwise, the user has to issue the refresh command explicitly to refresh Class Browser or Project Navigator. Selecting a component associated with the link change will also force the update.
- The number of user-defined relation types is no longer limited by the fixed set of relation colors. Instead, colors are now recycled among all relation types. Exceeding the number of supported relation types used to crash IDE.
- Automated IDE Error Reporting utility is fixed. Users are encouraged to send an error report every time they run into an IDE exception as the stack trace included in the error report can be a significant help in fixing the issue.
- Exception handling is modified to catch also non-UI thread exceptions. This should help catch previously unhandled exceptions.
- Exception reporting dialog is modified to include also ‘Contact Email’ and ‘Report Log’ fields.
- The class selection in Create Part dialog is revisited.
 - The separate name suggestion window is replaced with an appending completion feature.
 - Duplicate class names now include their library name in parenthesis, e.g. PARTS (LIB1).
 - Class list is now sorted.
- The 'Inherited' label in Component Editor was sometimes incorrectly set to a hyperlink and hovering mouse over the link caused IDE to crash. This issue is now fixed.
- Object naming is modified to allow names to start with a digit and end with a '?'.

- Pressing <space> on '[Create new attribute]' in Component Editor's Attribute List is modified to enter the edit mode allowing user to type the name for the new attribute.
- Error handling for 'Create Library' dialog is fixed to show a proper error message for invalid library names.
- Renaming an attribute in Component Editor's Attribute List is fixed to sort the attributes.
- Tooltip informing about an invalid object name is modified to match the enhanced naming convention. That is, an object name can also start with a digit and end with a '?'.
- Names ending with a '?' are restricted to only those objects that don't get stored in the file system as such. This is because '?' is considered a wild character in a pathname. As a result, only class, component, and attribute names can end with '?'.
- Since object names are not allowed to contain spaces, entering an object name is modified to either ignore the <space> key altogether or reject a name containing spaces.
- Renaming an attribute in Component Editor's Attribute List is modified to scroll to the renamed attribute automatically. Thus, the attribute remains visible after it has been renamed.
- The order of 'OK' and 'Cancel' buttons in the Preferences dialog is fixed to match that of the other dialogs.
- IDE's ability to adjust to increased default Windows text size is improved significantly. On Windows 7, you can change the default Windows text size with command: Right-click desktop > Screen resolution > Make text and other items larger or smaller.
- The problem of opening project files (functions, rules, structures, externals) for editing is fixed.
- Saving a project is modified to prompt the user to save also all modified project files currently open for editing.
- Event handling is improved to alleviate some of the hard-to-reproduce IDE crashes.
- Relation Browser's 'Expand All' command is optimized to retrieve all the data first and only then expand the graph. It used to expand the graph recursively one level at a time. Also, busy cursor is now shown during the expansion.
- At 125% font size, some dialogs were not rendered correctly. This is now fixed.
- Rule Editor's 'Indent Rule' command is disabled for read-only rules.
- Rule Editor's Copy command is modified to copy the whole read-only rule as selection is disabled.

- Updating of Navigator's Properties dialog data is fixed. Also, property titles are now capitalized.
- Query Analyzer search is fixed to handle string values correctly. Also, typos in attribute specification menu entries are fixed.
- Navigator's file category root nodes and Lisp file editor are unified to have the same New and Open file functionality.
- Component Editor's attribute origin display is fixed to be cleared when the current origin source goes away, for example, when it's deleted or its library/model is closed.
- Double-clicking Navigator's Projects and Models nodes is fixed to first check if the node is closed and, if it is, to expand the node. Only, if the node is already expanded, then double-clicking the node toggles between displaying all the projects/models and displaying just the current project/model.
- All dialogs now check if previously saved dialog location is viewable with the current monitor setting before reusing the location.
- Lisp File Editor's Expressions list box moved next to toolbar icons and Package list box is removed. Expressions list is fixed not to collect duplicate expressions. `defvar`, `defconstant`, and `defparameter` are now also collected.
- 'Edit Rule in Emacs' command is added to Attribute Editor's Rule Editor.
- Argument List Pane is added to Lisp File Editor for function and rule files. The Argument List Pane shares the space with the Output Pane and the two are toggled depending on which one has something to show.
- Lisp File Editor is enhanced to detect if the file currently being edited is also changed outside of IDE.
- Navigator's Lisp file node monitoring is improved.
- Navigator's Relations entries now have a right-click menu and default double-click functionality. The options are to open the relation definition file (default for double-click), start browsing relations, or open Relation Selection dialog.
- Create Model dialog now verifies that there isn't an identically named library already. Similarly, Create Library dialog checks for identically named models.
- 'Expand Subtree' command in various graph browsers is modified to expand just one level of the selected root node. I used to expand the whole graph.
- Navigator's 'Open All Libraries' command is optimized.
- New commands 'Save All Libraries', 'Revert All Libraries', and 'Close All Libraries' are added to Navigator's Libraries node.

- A new confirmation dialog is added to Lisp File Editor's save command.
- Lisp File Editor is enhanced to make a backup file before saving a file.
- Design rule compilation is fixed to handle Design++ reported Lisp reader (syntax) errors correctly.
- Exiting IDE when there is a modified, unsaved file open in Lisp File Editor is fixed to abort the exit if user cancels the save confirmation dialog
- 'View->Start Page' command is fixed to open the Start Page in a docked state.
- Print command in 'Print->Print Preview' dialog is modified to show printer selection.
- Graph printing is fixed to print only currently expanded graph.
- 'Locate in Navigator' command is added to object right-click menus in Library/Model/Relation Browsers and Component Editor.
- Class name right-click menu in Component Editor's header pane is fixed.
- Time stamps modified to use international format '1-Jan-2011 22:10:00'
- Term 'Toggle Facet Editor' is replaced with term 'Toggle Attribute Editor'.

Lisp/API

- Macro def-external-server is optimized and modified to allow the time-to-start argument to be also an expression.

C/API

- dppSystemIsUpdateDefault returned inverted answer.
- dppCommGetOpsysName worked only when dppChar=char.
- dppCommGetServerAddressName returned char instead of dppChar.
- dppComponentCreateFromFileAsync and dppComponentCreateFromFile incorrectly passed the filename as symbol, which mostly worked.
- Renamed dppPortAlloc to dppPortMalloc and added dppPortCalloc.
- Modified dppProjectFindObject to set return values to NULL if nothing is returned.
- Changed the array spec from *void to *dppStringArray in dppStringArrayGetValue and dppArrayAppendString.

- dppAttributeGetFacetListValueList argument facetnames can be NULL, in which case the default user facets will be returned.
- Fixed dppAttributeGetComment to ignore backup language.
- dppClassDeleteLink didn't handle error information correctly when error CAPI_SUBCLASS_DOES_NOT_EXIST was returned.
- Fixed dppComponentGetAttributeAndValueclassAndValueList to handle rule and locality flags correctly.
- dppComponentGetAttributeNameList returned invalid results.
- Fixed dppComponentGetAttributeAndFacetValueList, which didn't handle the attribute parsing correctly.
- Language keyword dppAttributeLANGUAGE_NONE removed as it was not ever implemented on the Lisp side.
- Fixed dppStringArraySort to work with NULL entries in stringArrays.
- Added test for dppModelCreate to check return value when library with same name exists already.

COM/API

- COM module names are renamed from "Bentley Design++ ..." to "Design++ ...".

.NET/API

- Namespace Bentley.dpp is renamed to DPI.DPP.

Starting Design⁺⁺ from CAD

- Modified to invoke the Lisp debugger if project or function specifications are done with invalid DPPSTARTUPPROJECT and DPPSTARTUPFUNCTION settings.

CAD Integration Manager (CIM)

- Fixed Symbol primitive attribute value parsing to correctly handle empty values.
- Modified Lisp, CIM and CAD code to correctly handle empty string values in GEO_LEVEL and GEO_COLOR attributes.

MicroStation Link

- Support for MicroStation V8i (SELECTseries 3) is added.
- AutoDrafter Symbol primitive is fixed to handle empty attribute values correctly.

- Design⁺⁺ MicroStation COM extap module is renamed from "Bentley Design⁺⁺ MicroStation COM Interface 1.0 Type Library" to "Design⁺⁺ MicroStation COM Interface 1.0 Type Library".

AutoCAD Link

- Support for AutoCAD 2012 and 2013 is added.
- With AutoCAD 2011 and older versions, acaddpp loads AutoCAD ribbon GUI explicitly. This can now be disabled by setting environment variable DPPLOADRIBBON to 'no'.
- Design⁺⁺ AutoCAD COM extap module is renamed from "Bentley Design⁺⁺ AutoCAD COM Interface 1.0 Type Library" to "Design⁺⁺ AutoCAD COM Interface 1.0 Type Library"
- The issue where a newly created PaperSpace viewport ends up inactive causing view related dimension creations to fail is fixed.
- Angular Dimensions witness line location calculation is fixed.
- Angular Dimensions continuation dimension points are now handled correctly.
- Viewport's Visual Style is now set to Hidden when AutoDrafter's VIEW_TYPE attribute is set to HIDDEN-LINE.
- The search for a seed file is extended to search all the normal locations in case the file is not found from the same directory as the drawing file itself. As a result, the search for a seed file is now similar to symbols search except Design⁺⁺ symbols directories are not searched.
- Lisp/API function dpp-cad-settings can now be used to change AutoCAD link settings. Previously only current values would be returned.
- Fixed AutoCAD path fetch to work even when multiple versions of AutoCAD, like 2011 and 2012, are installed.

AutoDrafter

- See AutoCAD section above for some AutoCAD specific modifications.

Emacs Interface

- Updated to Emacs 23.4

Java/API

- Compiled with JDK 1.7.4

Installation

- When 'Configure' option is not shown, the 'Client Configuration' option description was shown incorrectly as the 'Configure' description. This is now fixed.

Licensing

- SELECT Licensing is replaced with CrypKey, which now has much improved network licensing support.
 - Independent Server Setup - Any workstation on the company network can be configured as a license server without installing Design⁺⁺ to it.
 - True Network Licensing - Design⁺⁺ finds its license anywhere on the company network automatically. And, there is no need to share server directories.
 - Redundant License Servers - If one license server goes offline, Design⁺⁺ can automatically find and switch to another server, on the fly.
 - Network License Control - License server can specify which domains, workstations, or users are included or excluded from its Design⁺⁺ license. Likewise, client workstation can specify which license servers Design⁺⁺ can obtain its license from.
- Lisp/API function dpp-license-status is modified to return Design⁺⁺ license status instead of opening the License Management Tool. It returns 2 keywords. The 1st keyword is the actual license status :OK, :EVALUATION, :ACADEMIC, or NIL and the 2nd keyword is the license type :STANDALONE, :NETWORK-CLIENT, or :NETWORK-SERVER. The License Management Tool can be opened with another Lisp/API function dpp-license-manager.

Miscellaneous

- Document for Vessel Exercise tutorial is updated.