

October 27, 2008

Bentley Design⁺⁺ V8i Release Notes

Welcome to Bentley Design⁺⁺ V8i. This is the second major Design⁺⁺ release since Bentley Systems acquired Design Power, Inc. Design⁺⁺ V8i is a part of Bentley's new V8i software portfolio for infrastructure professionals.

Design⁺⁺ V8i is supported on Windows XP, Vista, and Server 2003/2008 platforms. Windows 2000 is no longer supported.

Highlights

Priority-Based Change Management

Design⁺⁺ change management has been optimized significantly by maintaining initial attribute calculation order to prioritize recalculations. This approach eliminates virtually all extra attribute recalculations during change propagation. As there are no extra recalculations, changes to model structure can be applied immediately.

Updated Installation

Design⁺⁺ now comes with separate installs for Development and Deployment versions. The benefit is a much reduced Deployment install size of only 30 MB, whereas the Development install is 130 MB.

Design⁺⁺ install has been updated to use InstallShield 2009 and has a new updated look and feel.

Improved Integration with SELECT Licensing

Design⁺⁺ licensing has been modified to allow Design⁺⁺ Deployment version to run also with a Design⁺⁺ Development-only license. This makes it easier for developers to test and package their applications for deployment.

Design⁺⁺ is now available through Bentley's academic licensing program, BE Careers Network.

VBA Support for MicroStation and AutoCAD

Design⁺⁺ has a new geometric free primitive type supporting VBA. This allows free primitives to be implemented directly in the standard CAD VBA environment.

VBA code can also be executed directly from design rules and Lisp functions through a new Lisp/API function `dpp-cad-execute-vba`. This requires no additional coding on the CAD side, for example: `(dpp-cad-execute-vba "ThisDrawing.Save")`

Design⁺⁺ example project `geo-test` contains a sample VBA project for both MicroStation and AutoCAD. The sample project implements several VBA free primitives and demonstrates other use of VBA.

Support for MicroStation V8i Added

Design⁺⁺ MicroStation link now supports MicroStation V8i and XM Edition through the native code MDL link. MicroStation V7 (aka J) is still supported through the old, non-native MDL link. Support for MicroStation 2004 has been dropped.

Support for AutoCAD 2009 Added

Design⁺⁺ AutoCAD link now supports AutoCAD 2009. Support for older AutoCAD versions 2000, 2000i, and 2002 has been dropped.

Supported CAD Versions

Design⁺⁺ V8i integrates with the following CAD versions.

AutoCAD: 2004, 2005, 2006, 2007, 2008, and 2009

MicroStation: V7 (aka J), V8 XM Edition, and V8i

Visio: 2000, 2000 SR1, 2002, 2003, and 2007

Installation

- Design⁺⁺ now comes with separate installs for Development and Deployment versions. The benefit is a much reduced Deployment install size of only 30 MB, whereas the Development install is 130 MB.
- Design⁺⁺ install has been updated to use InstallShield 2009 and has a new updated look and feel.
- Design⁺⁺ no longer includes a copy of Java Runtime Environment (JRE) in <d++>/java directory, instead a standard JRE is installed during installation if an existing version is not found.

License Manager

- Design⁺⁺ is now available through Bentley's academic licensing program, BE Careers Network.
- Design⁺⁺ licensing has been modified to allow Design⁺⁺ Deployment version to run also with a Design⁺⁺ Development-only license. This makes it easier for developers to test and package their applications for deployment.
- SELECT license client has been updated to the latest V8i version.

Design⁺⁺ Core

- Priority Based Change Management

Change management has been optimized significantly by maintaining initial attribute calculation order to prioritize recalculations. This approach eliminates virtually all extra attribute recalculations during change propagation. As there are no extra recalculations, changes to model structure can be applied immediately.

- Valueclass for the GEO_TYPE attribute fixed. The valueclass contained old geometric types not compatible with those supported by the current CAD Integration Manager. It was also missing newer types, like LINEAR_DIMENSION and XREF.
- Installation of dot notation based component and attribute referencing syntax fixed. This allows dot notation based expressions to be executed in the Command Interpreter just like normal design rule expressions. You can test this by evaluating '?P in the Command Interpreter. The result should be (:? P).
- ReportWriter's starter function dpp-rg-gui fixed to correctly reload ReportWriter GUIB definition files in case GUIB was restarted after ReportWriter was already in use.
- Change propagation fixed to correctly handle components that are initially deleted and later re-created all while delayed for redetermination.
- Change propagation fixed to establish proper dependencies among a chain of depending attributes even if some of them are initially determined to be NIL. This assures that those attributes that are determined to have a valid value (non-NIL) are properly redetermined later.
- Sending geometric data to CAD has been modified to traps Lisp printer errors. This prevents insufficient geometric data from being sent to CAD. In case of an error, the user can either proceed to the next component or cancel the 'Show Geometry' request and automatically jump editing the failed component (assuming UIP is running).
- Franz ACL SMTP module is now built into Design⁺⁺. Among other things, it allows email messages to be sent directly from Lisp.
- Design⁺⁺ now runs on top of the latest Franz ACL 8.1.

Design Rule Language

- Special dot notation keyword PARENT has been renamed to ASSEMBLY to better match the current naming convention. Here are some examples of dot notation expressions using the new keyword.

```
?assembly      -> (:assembly self)
?comp.assembly -> (:assembly comp)
?assembly.attribute -> (:? (:assembly self) attribute)
```

Note the following side effects of removing the PARENT keyword.

```
?parent        -> (:? parent)
?comp.parent   -> (:? comp parent)
?parent()      -> Error
?.parent       -> (:? self parent)
?.parent()     -> (parent self)
?comp.parent() -> (parent comp)
```

- Self-Guided Vessel Exercise tutorial has been upgraded to use the new dot notation based design rule referencing syntax.

Developer's User Interface (UIP)

- A new command 'Example Projects...' has been added to the main dialog's Help menu to improve access to example projects.
- A new command 'Tutorial' has been added to the main dialog's Help menu to improve access to Vessel Exercise tutorial.
- A new command 'Geometry>Highlight in CAD' has been added to Model Editor's Model menu.
- A new command 'Report...' has been added to the main dialog's model right-click menu.
- The 'Report...' command has been repositioned to be in the same position in all the menus.
- The geometry related commands have been placed to the bottom of Component Editor's Model menu.
- Project directory browser doesn't hide hidden directories anymore to allow access to default project directory path.
- Component Editor fixed to remember 'Show Realnames' settings when facet display settings are changed.
- Modified to show an error dialog if program resources can't be loaded when UIP is started.

User Interface Server (UIS)

- File/Directory selector has been modified to show hidden files and directories.
- Modified to show an error message when receiving an invalid message from Design⁺⁺ Core (Lisp).

Design Rule Editor (DRE)

- Context Sensitive Menu modified to make sure that the sub menu is cleared if no main menu entry was selected.
- Commands in the 'Edit>Delete' submenu have been renamed by removing the word 'Delete' from each of them.
- 'Edit>Indent' menu has been reorganized.
- Handling of rules with lots of tabulator characters has been improved. Opening such rules caused DRE to crash occasionally.
- An indentation problem, which sometimes added a lot of extra spaces to the beginning of a line, has been fixed.

C/API

- An error in parsing component path info in C/API function `dppModelGetComponentList` has been fixed.
- Strtok related functions, `dppFirstToken` and `dppNextToken`, have been removed from portability manager. If you have used either of these functions, you can probably replace them with `strtok` or similar.
- New C/API functions in this release are `dppStringLispify` and `dppModelHighlightComponent`

COM/API

- A new registration-free COM/API server module introduced. The new module, `dppCOMapi.dll`, implements the `dppComm` functionality of `dppCOMserver` as a DLL. In practice this means that `dppCOMapi` does not need to be registered and, as a result, multiple versions of it can be safely used simultaneously. As most COM/API clients are using only the `dppComm` functionality, this initial version does not implement the rest of `dppCOMserver` functionality.

CAD Integration Manager (CIM)

- Error in symbol primitive's `modelScale` argument usage fixed.
- Hidden line view is no longer supported.
- A new CIM function `dppCadLispify` added.
- New VBA code execution support added. This allows VBA code to be executed directly from design rules and Lisp functions through a new Lisp/API function `dpp-cad-execute-vba`. This requires no additional coding on the CAD side, for example: (`dpp-cad-execute-vba "ThisDrawing.Save"`)

Design⁺⁺ example project `geo-test` contains a sample VBA project for both MicroStation and AutoCAD. The sample project implements several VBA free primitives and demonstrates other use of VBA.

MicroStation Link

- Design⁺⁺ MicroStation link supports MicroStation V8i and XM Edition through the native code MDL link.
- MicroStation V7 (aka J) is still supported through the old, non-native MDL link
- Support for MicroStation 2004 and non-native XM Edition have been dropped.
- Design⁺⁺ has a new geometric free primitive type supporting VBA. This allows free primitives to be implemented directly in the standard CAD VBA environment.

VBA code can also be executed directly from design rules and Lisp functions through a new Lisp/API function `dpp-cad-execute-vba`. This requires no additional coding on the CAD side, for example: (`dpp-cad-execute-vba "ThisDrawing.Save"`)

Design⁺⁺ example project geo-test contains a sample VBA project for both MicroStation and AutoCAD. The sample project implements several VBA free primitives and demonstrates other use of VBA.

- Naming of Autocalc related functions has been changed from dppExtapAutocalc* to dppExtap*Autocalc to better match with other CAD API functions.
- Hidden line view is no longer supported.
- Native code MDL link returns ElementID for primitives instead of component name as was the case with previous link versions.
- Modified to verify that a primitive's location is in the current design plane.
- Modified to use subunits in AutoDrafter by default.
- A new Lisp/API function dpp-ms-execute-keyin added
- A new user function dppExtapLispify added.
- User function autoDrafter_exportDrawing has been removed.

AutoCAD Link

- AutoCAD link now supports AutoCAD 2009.
- Support for older AutoCAD versions 2000, 2000i, and 2002 has been dropped.
- Design⁺⁺ has a new geometric free primitive type supporting VBA. This allows free primitives to be implemented directly in the standard CAD VBA environment.

VBA code can also be executed directly from design rules and Lisp functions through a new Lisp/API function dpp-cad-execute-vba. This requires no additional coding on the CAD side, for example: (dpp-cad-execute-vba "ThisDrawing.Save")

Design⁺⁺ example project geo-test contains a sample VBA project for both MicroStation and AutoCAD. The sample project implements several VBA free primitives and demonstrates other use of VBA.

- A new user function dppExtapLispify added.
- Error messages for invalid colors and layers has been improved.
- Hidden line view is no longer supported.
- AutoDrafter centerline symbol is now implemented as code "\U+2104" in style ADESK1 instead of character g in AMGDT font.

Database Link

- Support for old Oracle link has been removed, use ODBC link instead.

User Interface Builder (GUIB)

- Lisp/API function `dpp-gui-post-event-dialog` has been fixed to work asynchronously and not to receive an unwanted answer message from GUIB.

Emacs Interface

- Emacs version has been updated to GNU Emacs 22.3.
- XEmacs is no longer supported.

Miscellaneous

- Names for example projects have been simplified and the naming separator has been switched to from `'_'` to `'-'`.
- Example projects Plant and Bike have been upgraded by removing obsolete RDB link and reporting related code.