

Autodesk® Inventor® Automation Professional and Runtime 2010 Release Notes, Update 6, build 3.5.473

Product Release Date: February 1, 2010 This document last updated: February 1, 2010

For the very latest update of the Autodesk Inventor Automation software and any hot fixes or service packs, please visit the Automation product page here:

<http://usa.autodesk.com/adsk/servlet/pc/index?id=8057864&siteID=123112>. Click on the Data & Downloads link and then on the Updates & Service Packs link.

Release Notes - Readme file Autodesk, Inc. This file contains information that became available since the release of Autodesk Inventor Automation 2010 that you will find useful while using Inventor Automation Professional or Runtime. Please obtain the latest version of this document for the release it covers – it will be continually updated as new information comes to light.

This document is broken into four sections:

- Section 1: Installation
- Section 2: What is New in Intent 2010
- Section 3: Changes
- Section 4: Known Issues

General Note on Naming

The official names of the products described in this document are Autodesk Inventor Automation Professional 2010 and Autodesk Inventor Automation Runtime 2010. However, rather than use these rather long names, this document uses the common moniker “Intent 2010” for both products, with Professional or Runtime called out as necessary. It is necessary to use the official names during licensing, purchasing, and support activity to ensure that a precise release is described.

1 Installation

1.1 Compatibility

Intent 2010 is supported on the following operating systems:

- 32-Bit Windows XP
- 32-Bit Windows Vista
- 64-Bit Windows XP
- 64-Bit Windows Vista

Intent 2010 does not require any CAD host to be installed. However, it will install bindings for the following hosts if they are present:

- Autodesk Inventor 2009
- Autodesk Inventor 2010
- AutoCAD 2010 (English and Japanese only)
- AutoCAD Mechanical 2010 (English and Japanese only)

1.2 Inventor 2009 Hot fixes, Service Packs, and Samples

If Intent 2010 is to be installed for use with Inventor 2009, Autodesk Inventor 2009 Service Pack 1 should be installed prior to installing Intent 2010. Service Pack 1 includes the following required hotfixes:

TS1082766 - HOTFIX: Unable to locate model geometry (Face) curves in cut view

TS1082767 - HOTFIX: Missing critical OnNewView (after) event

Download Autodesk Inventor 2009 Service Pack 1 from the Autodesk web site ([Autodesk Inventor 2009 Service Pack 1](#)).

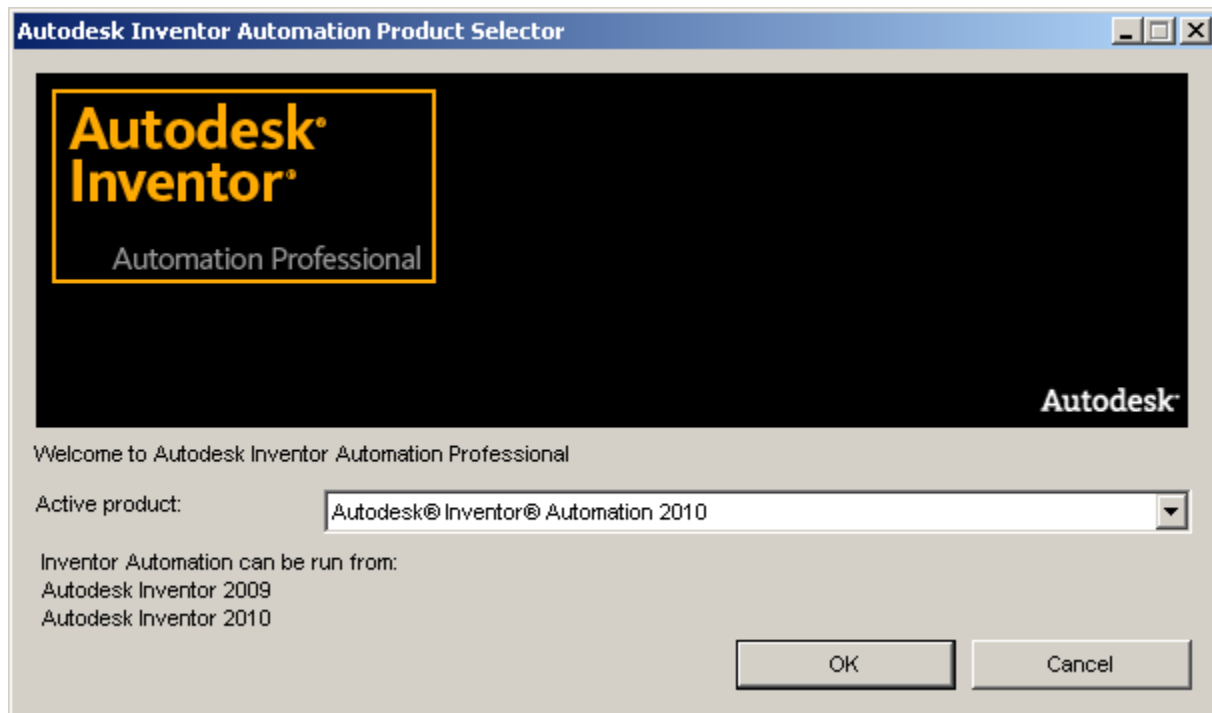
The Intent Samples for Inventor 2009 are not installed by the installer, but they are included with the installation package (for the Professional product only). After installation, they can be found in a self-extracting package located at:

C:\Autodesk\Automation_Professional_2010_US_Win_32bit\Samples\Samples.exe

1.3 Running Intent 2010

Intent 2010 is a full release. It can be installed simultaneously with prior major releases of Intent, although host bindings may need to be adjusted if it is desired to run Inventor Automation Professional 2009 with Inventor 2009. Minor releases cannot be installed simultaneously.

If you need to switch between the 2009 and 2010 Intent releases, you can find a Product Selector utility on the Start Menu; see [Start]\Programs\Autodesk\Inventor Automation Professional 2010\Inventor Addin Settings.



Select the active product and click on OK button. Be aware that this action will not change the references to the Intent assemblies in your custom projects, you will need to make the change manually and re-compile.

1.4 Installation Requirements

To install Intent 2010 you must have the appropriate self-extracting installer file for your platform (32 or 64 bit) and variant (Professional or Runtime). The file names are self-explanatory. Only one of these may be installed:

- Autodesk_Inventor_Automation_Professional_2010_English_Win_32bit.exe
- Autodesk_Inventor_Automation_Professional_2010_English_Win_64bit.exe
- Autodesk_Inventor_Automation_Runtime_2010_English_Win_32bit.exe
- Autodesk_Inventor_Automation_Runtime_2010_English_Win_64bit.exe

1.5 Host Preference and Priority

The installer will install bindings for hosts as follows.

If Inventor 2010 is present, it will be set to use Intent 2010.

If Inventor 2009 is present, it will be set to use Intent 2010.

If plain AutoCAD/US-English is present, install to it

Else If AutoCAD Mech/ US-English is present, install to it

Else If plain AutoCAD/Japanese is present, install to it

Else If AutoCAD Mech/Japanese is present, install to it

Else (no known autocad is present), do not install AutoCAD binding.

Only one AutoCAD binding will be set no matter how different versions of AutoCAD are installed.

If none of the supported hosts are present, Intent 2010 will still install its core modules and the command line and UI versions of MinHost.exe, a simple application that can be used to evaluate rules which have no geometric modeling requirement. See the section on MinHost under *What is New in Intent 2010* for more details.

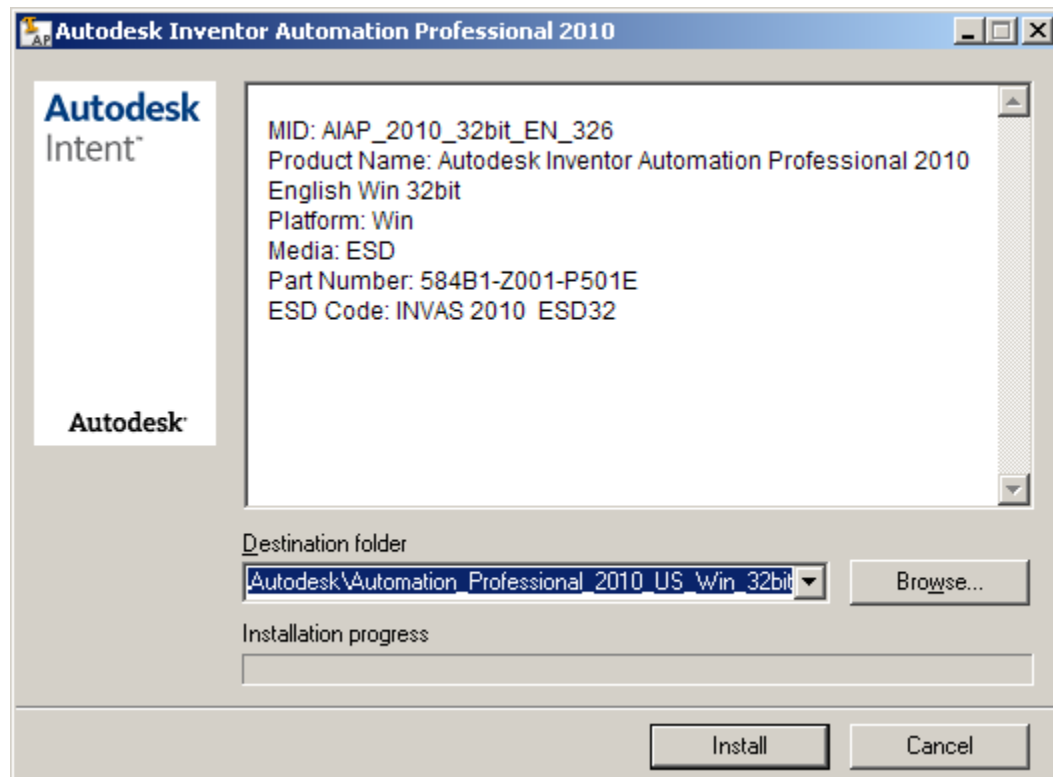
1.6 License Method Matching

Intent 2010 will only run properly on 2010 hosts with matching license methods. That is, if Intent 2010 is to be run with Inventor 2010, the license method (either network or stand alone) must be the same. The same applies to AutoCAD 2010 or AutoCAD Mechanical 2010: the license method must match.

The license method does not need to match with Inventor 2009, since it uses an older licensing technology.

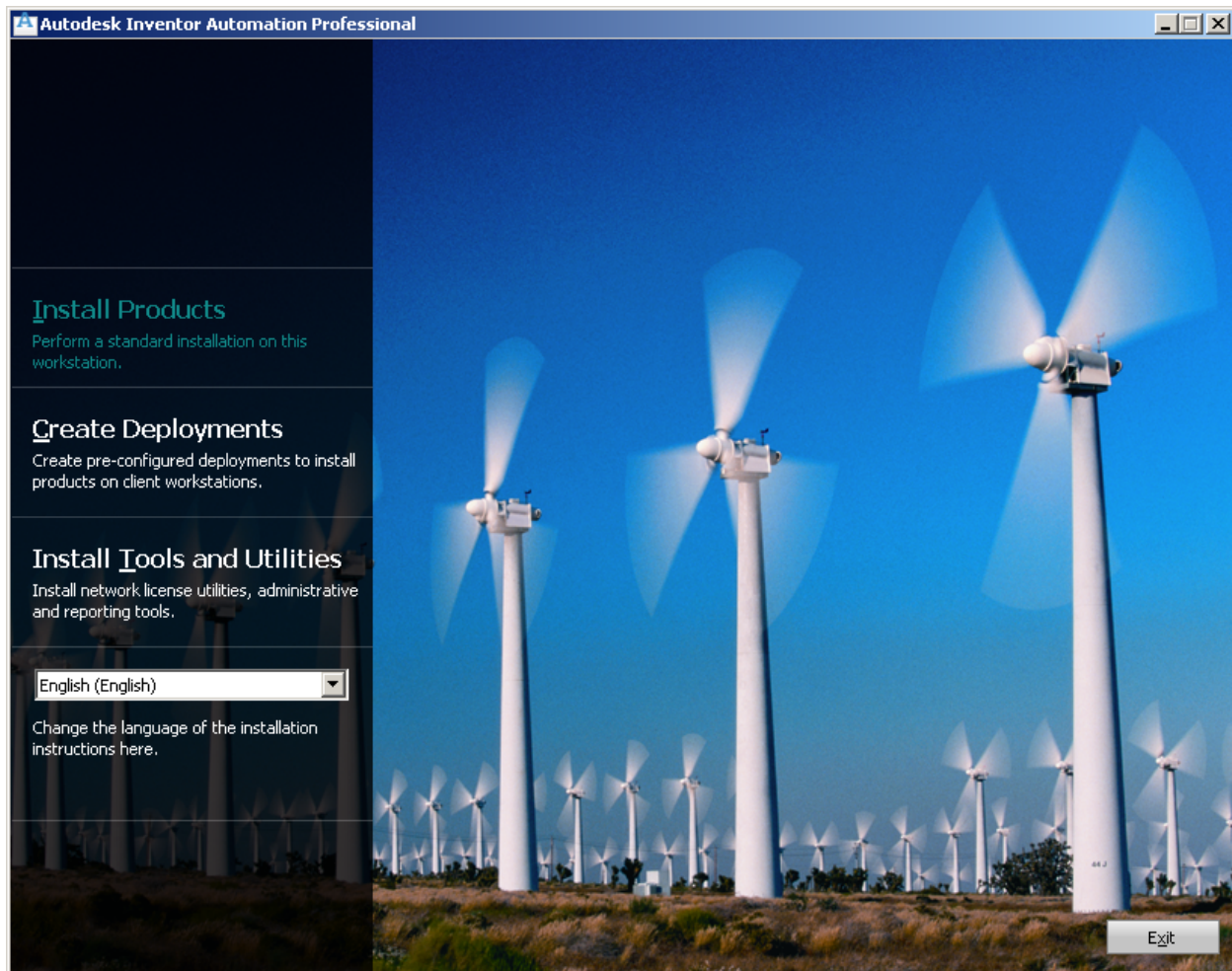
1.7 Installation Steps

To install Intent 2010, run one of the exe files in section 1.4. The following dialog will appear:



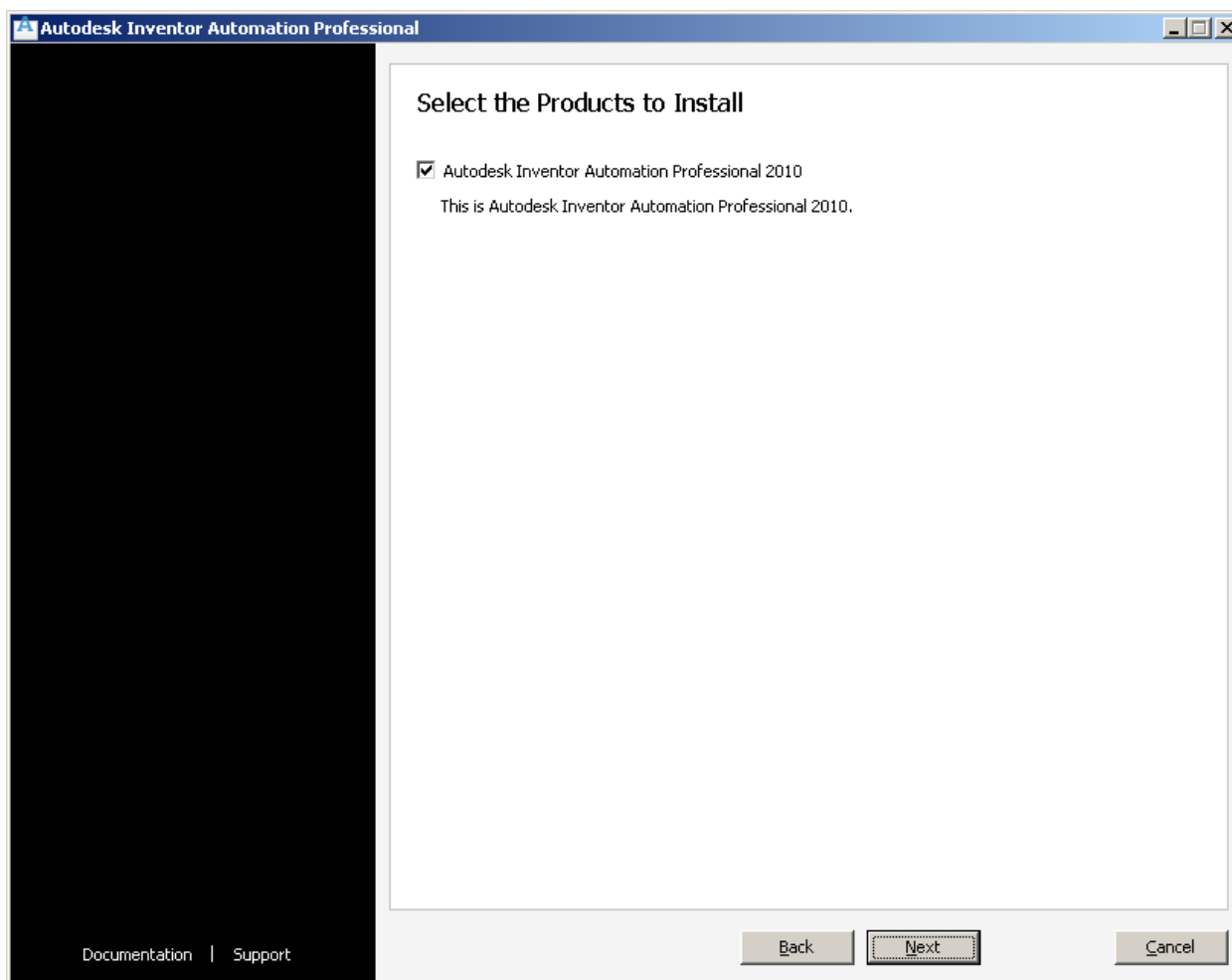
Click Install to proceed.

The Launch Dialog will appear:



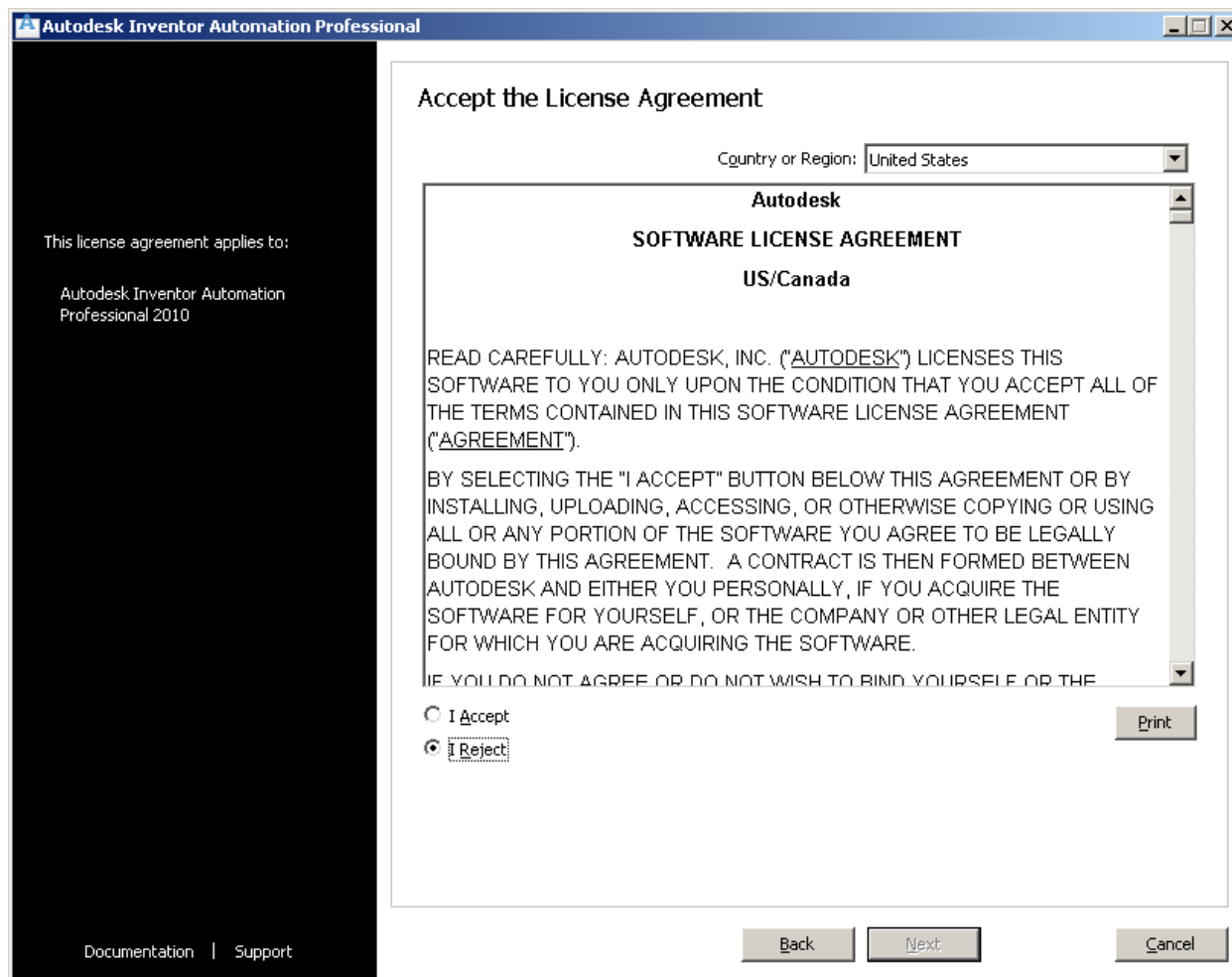
Select Install Products.

The Product Selection Dialog will appear.



Click on Next.

The License Dialog will appear. You must accept the license to continue installation.



After accepting, click Next.

The User Information Dialog will appear. Enter your serial number and product key (obtained from Autodesk or a channel partner). Verify your personal information. Note that you can supply 000-00000000 for the serial number to get a single 30-day trial license. This trial can only be performed once for a given computer, and only for stand alone licensing.

The information you enter here is permanent; it is available in the About box under the help menu item within the product.

Documentation | Support

Product and User Information

Serial number*:
 -

Product key:

First name:

Last name:

Organization:

*If you have not yet purchased Autodesk Inventor Automation Professional, enter 000-00000000 as the serial number, and 00000 as the product key. You can purchase Autodesk Inventor Automation Professional later.

Please note that all data gathered during product activation is used in accordance with the Autodesk [Privacy Policy](#)

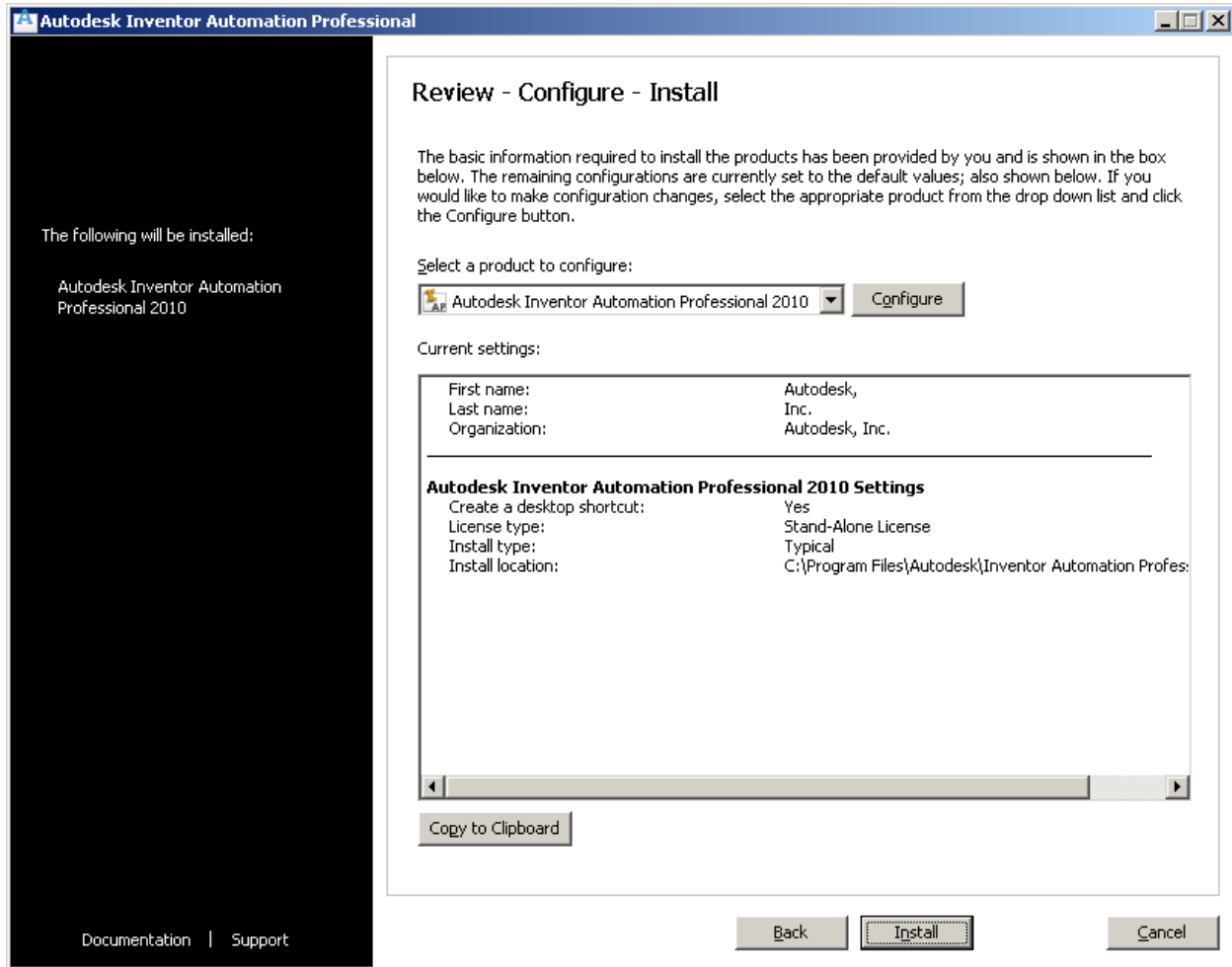
Back Next Cancel

The Product Key is an identifier for the product. Use one of the following:

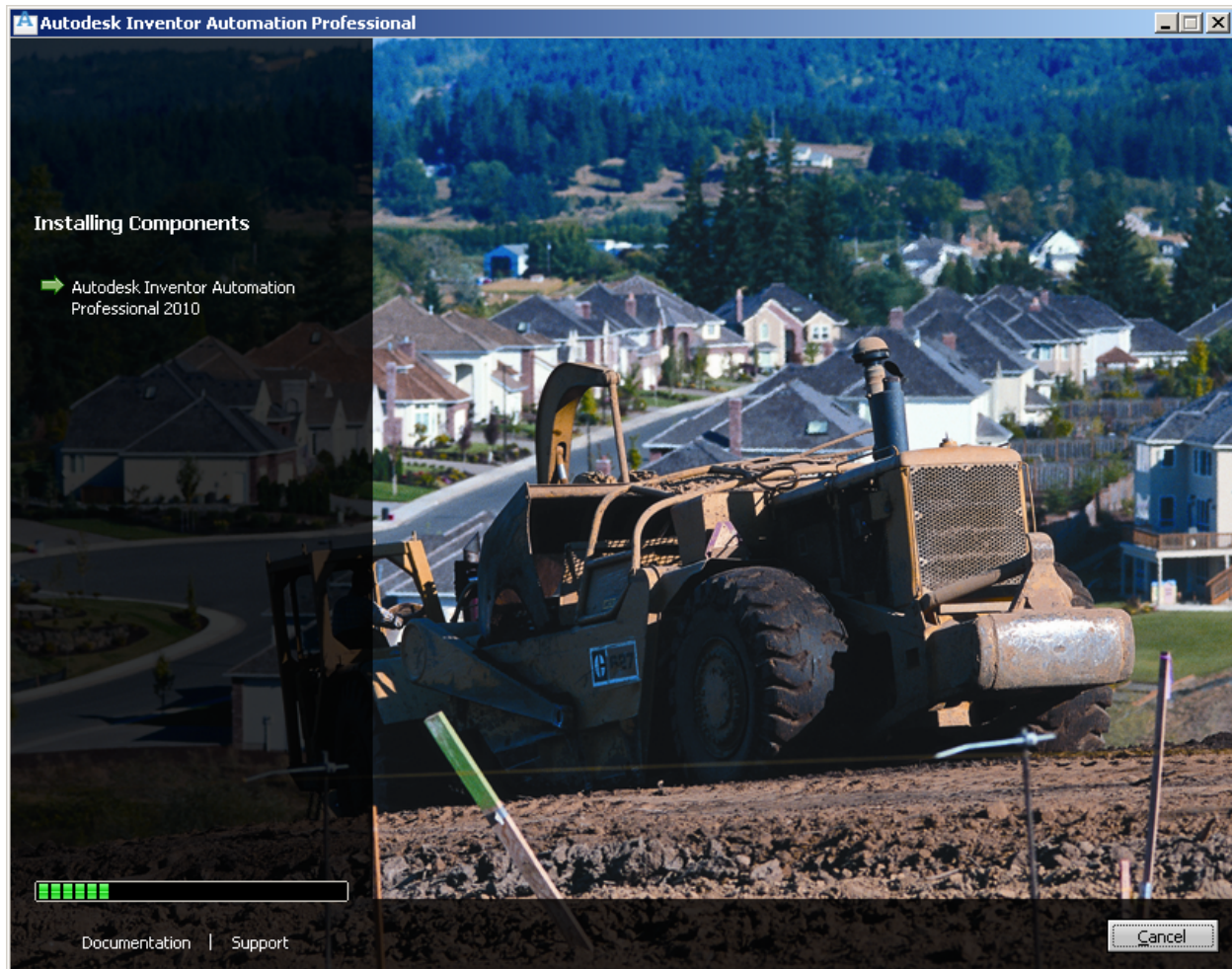
- 584B1 for the Professional product
- 636B1 for the Runtime product

Once complete, click Next.

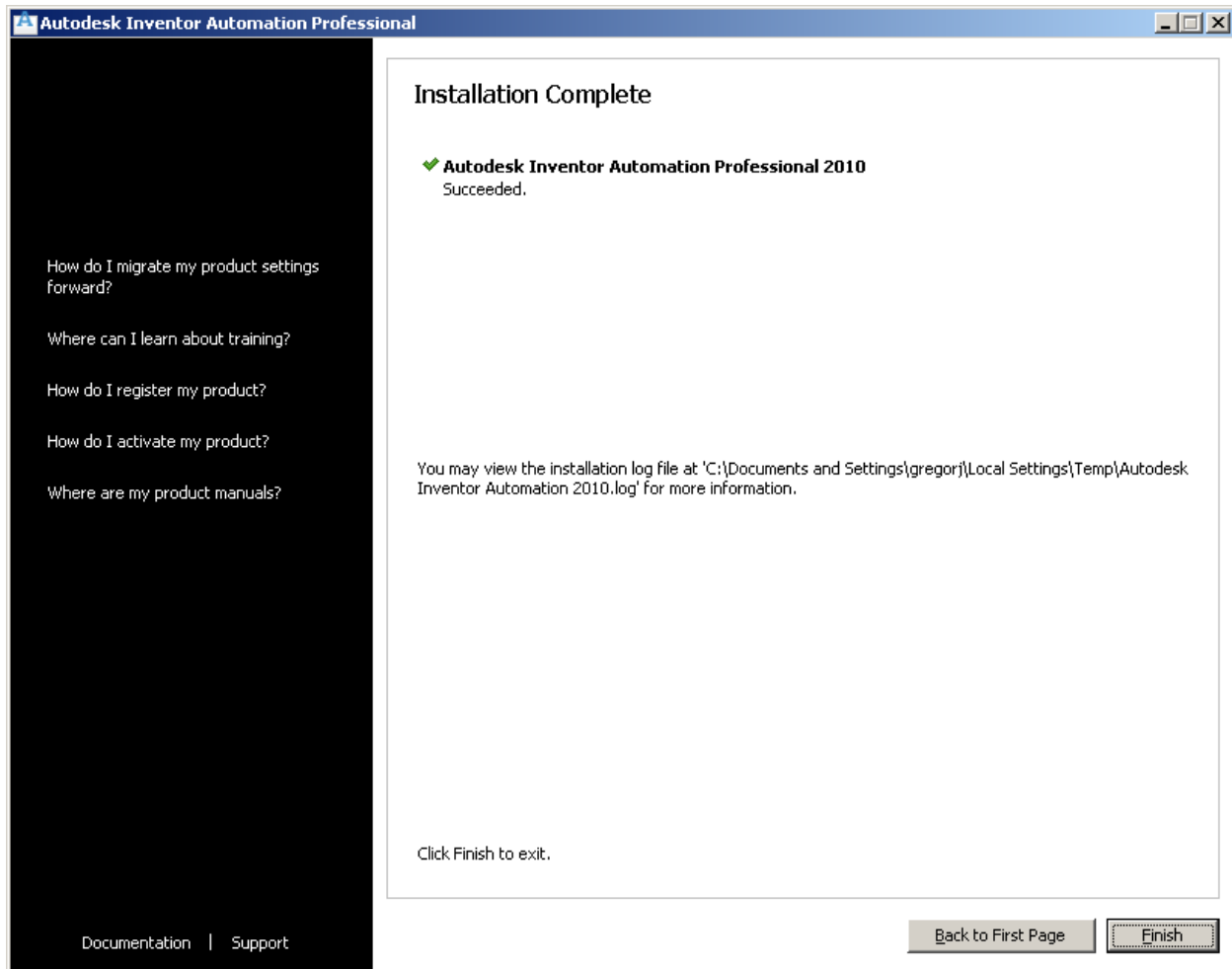
The Confirm Setup Dialog will appear. Note that a Standalone (not Network) installation is assumed. You must change this if your serial number represents a network license. Select Configure to change the default installation behavior.



Once configuration is completed, the installation will begin.



Once installation is completed, the Install Complete Dialog will appear.



Click Finish to complete installation.

2 What is New in Intent 2010

Intent 2010 is a new full release, and contains several new features, as well as defect fixes.

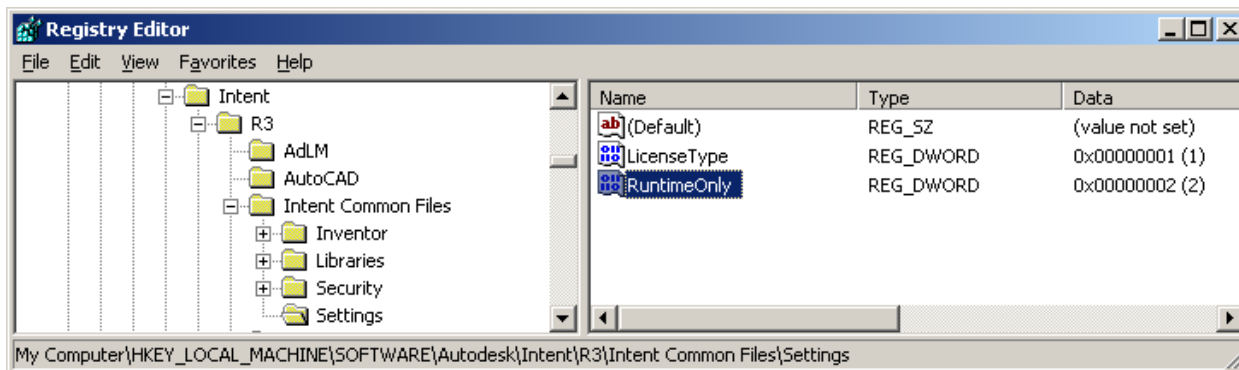
2.1 Runtime

Intent 2010 is delivered in two variants. The Professional variant provides full rule authoring capability. The Runtime variant allows deployed applications to have smaller installation footprints, restricted rule viewing, editing, and manipulation options, and a lower cost.

Runtime Mode in Professional

Intent Runtime 2010 is separately licensed and installed; Runtime licenses cannot be used for Professional authorization, but a Professional license does allow the developer to start in “Runtime mode” to observe the effect of running with the Runtime restrictions. In order to perform this switch, the user needs to edit the system registry (and have rights to do so).

In order to simulate Runtime, the registry entry
HKEY_LOCAL_MACHINE\SOFTWARE\Autodesk\Intent\R3\Intent Common
Files\Settings\RuntimeOnly needs to be set to a DWORD value of 2.



Developers wishing to create installers for deployments of Intent Runtime should contact Intent.Support@autodesk.com for assistance. The necessary merge modules are not included in the Professional installation.

2.2 Network Licensing

Intent 2010 supports Network or “floating” licensing. This type of license is maintained on a server or servers, and licenses are checked out over a local area network. There are a few restrictions:

1. When Intent 2010 is hosted by another Autodesk “2010” series product, the license methods (network or standalone) must match. For example, a network-licensed Intent 2010 can only be used on a network-licensed Inventor 2010. This restriction does not apply to Inventor 2009, because the licensing technologies are different.
2. Network licensing only works over a LAN or a VPN. It will *not* work across the internet without a VPN or similar technology.
3. The setup of network license servers is beyond the scope of this document. Because of the restriction on similarly-licensed hosts, it is expected that system administrators for this type of installation will be familiar with the setup process. If a network-licensed Intent 2010 process cannot find the license server, it will ask the user to resolve the issue.

2.3 .Net Framework Support

The Intent Language now supports .Net data types natively in the language. This makes it much easier to integrate external functions into Intent.

The key change in the language is the generalization of the “dot” operator from a purely rule-referencing operator into a more general “member” dereferencing operator. Intent now handles type names differently; native Intent types still exist, such as Number, but now a rule can have a .Net type, such as System.DateTime.

There are some restrictions to this new capability.

2.3.1 Literals

Literals always return their original native Intent data type. Where Intent native types and .Net data types are identical, they are generally automatically converted. However, there is little automatic conversion in other cases. For example, .Net enumerated types cannot be used as Integer values in Intent. Effective with the 2010 Update 1 release, some automatic conversions have been removed because they prevented proper argument passing to and from .Net calls. Going forward, Intent will only automatically convert when it is possible to convert without loss of fidelity in both directions, such as for String and System::String. All other conversions will require explicit conversion functions.

2.3.2 Full Specification

Currently, .Net types must be fully specified. There is no “using” or “imports” statement.

2.3.3 GAC-resident assemblies must be loaded

Currently, Intent will search only two places for assemblies to resolve type names: the current application’s loaded assemblies, and the Intent search path. This means that unless an assembly is loaded by the host process, or resides in (or is loaded by) a DLL in the Intent search path, it will not be found. The GAC is not searched. This restriction will be removed in a future release.

A workaround is to create a custom .Net assembly DLL which loads the needed assemblies, and place that in the Intent search path, and ensure it gets loaded.

2.3.4 IntentAPI is Restricted

Calls to the IntentAPI are restricted; any call that will require evaluation of the current Intent model will result in a re-entrancy error. In addition, calls to change the current model, such as a add a rule to a loaded Design, are also not permitted. However, calls to non-evaluation aspects of the API are available. This restriction may be removed in a future release, although it is likely that there will always be some operations which are not permitted during evaluation.

2.3.5 Calling Conventions

Not all calling conventions supported by .Net are permitted in Intent. For example, “byRef” arguments are not supported. It is not possible to return values via function arguments. Optional arguments, as defined by Intent, are not supported for .Net calls.

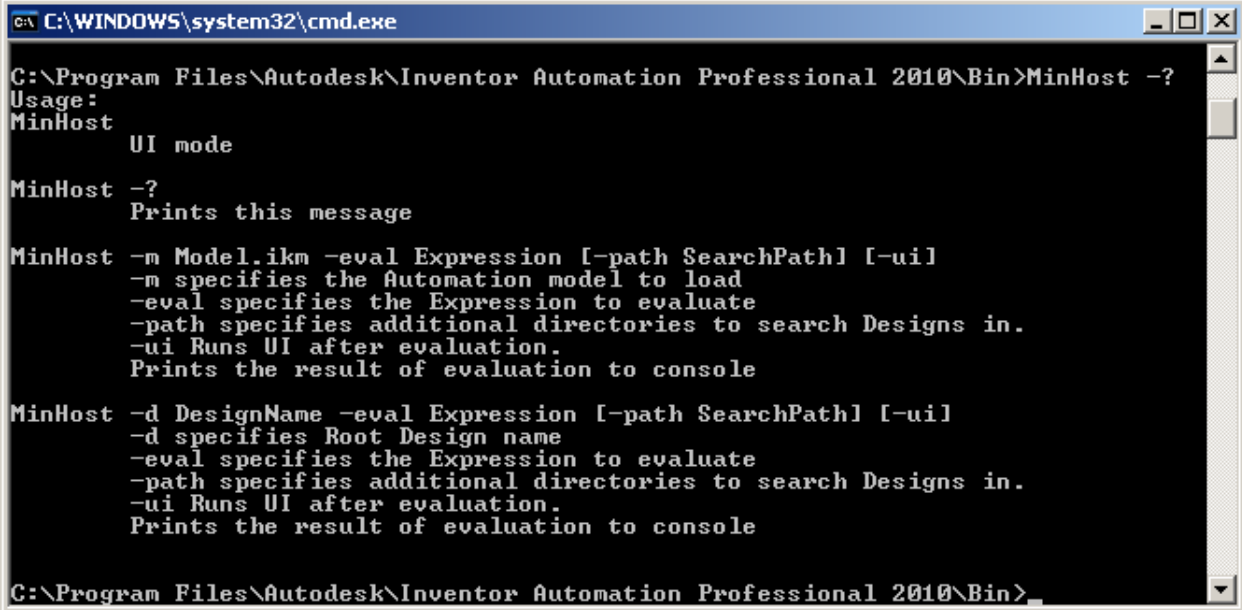
2.4 64-bit Support

Intent is now fully supported on 64-bit platforms.

2.5 New MinHost Demo Application

With Intent 2010, Autodesk is supplying a demo verison of a new “unhosted” implementation of Intent which can be used for evaluation and testing of generic rule sets, command-line rule evaluation, and other applications which do not need a geometric modeler “host”. Called MinHost, this implementation of Intent is a minimal collection of all the essential ingredients to make a full Intent implementation.

When run from the command line, MinHost can evaluate rules, load or save .IKM model files, or perform any action which can be invoked via rules.

A screenshot of a Windows command prompt window. The title bar reads "C:\WINDOWS\system32\cmd.exe". The command prompt shows the execution of "MinHost -?" which displays the following usage information:

```
C:\Program Files\Autodesk\Inventor Automation Professional 2010\Bin>MinHost -?
Usage:
MinHost
    UI mode

MinHost -?
    Prints this message

MinHost -m Model.ikm -eval Expression [-path SearchPath] [-ui]
    -m specifies the Automation model to load
    -eval specifies the Expression to evaluate
    -path specifies additional directories to search Designs in.
    -ui Runs UI after evaluation.
    Prints the result of evaluation to console

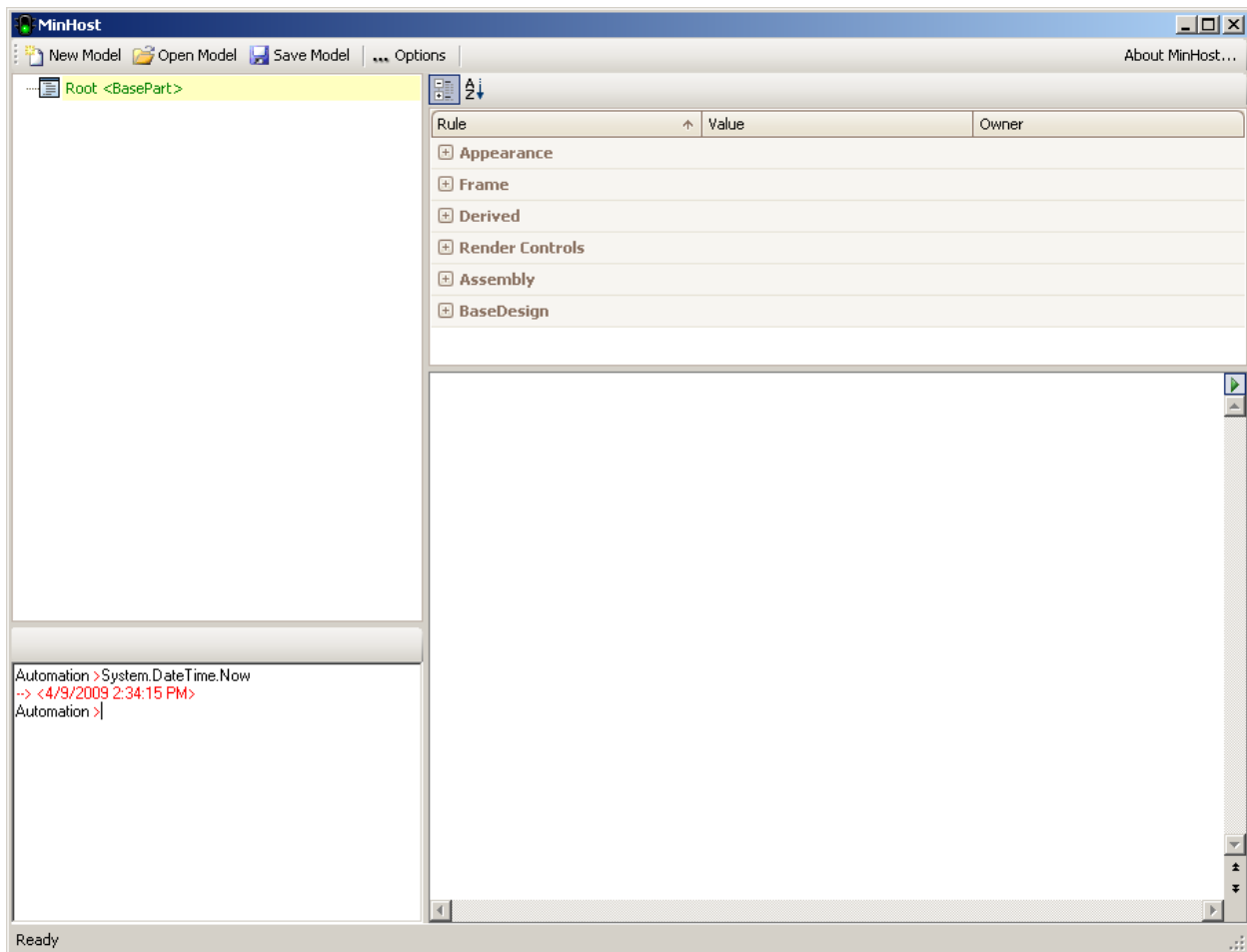
MinHost -d DesignName -eval Expression [-path SearchPath] [-ui]
    -d specifies Root Design name
    -eval specifies the Expression to evaluate
    -path specifies additional directories to search Designs in.
    -ui Runs UI after evaluation.
    Prints the result of evaluation to console

C:\Program Files\Autodesk\Inventor Automation Professional 2010\Bin>
```

The intention of a command-line Intent system is to support batch operations, “back office” systems, or other non-interactive applications. Note that the command line access does “one” evaluation and then returns; it does not support multiple commands. There is no “Intent command loop” which can accept directives to Intent itself.

This form of MinHost is specifically well-suited for test applications. Although it cannot be used to test rules which require a CAD host to run, it can be used to test generic rules which sometimes form a large portion of Intent applications. All standard Intent libraries are included.

When run as an interactive application, all of the same user-interface features that are present in the other hosted versions are present, such as the immediate window, the parameters and properties panes, and other elements. It does not, however, include the Design Editor, although a simple Dynamic Rule Editor is included.



One important feature of MinHost is that it does not operate like CAD-hosted versions of Intent: it does not reference the RenderSelf rule after changes are made.

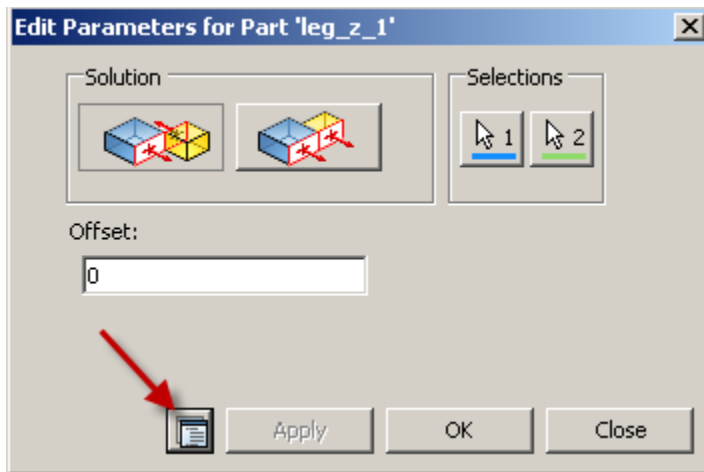
2.6 IKB Files now common and supported across platforms

IKB files now work with both x64 and x86 architectures. The same IKB file can be used on either platform, and can be created on either platform.

2.7 Customizable Part Editors

The Part Editor, which appears during an Add Child or Edit Parameters operation, is now a customizable element of the user interface. Intent Developers can now develop their own Part Editors which appear on a per-Design basis. These editors must be developed using a Microsoft .Net Framework compatible tool, and are not defined by Intent rules.

This new feature allows developers to add considerable custom content and appearance to their applications without having to write an entire application framework to support it.



Note that the arrow highlights a button that allows the user to return to the default Part Editor grid-style behavior.

2.8 New Features for Inventor

There are several new features in the Inventor 2010 integration.

2.8.1 Dimensions: Centermarks, Centerlines, Bisectors

Intent 2010 supports adoption of Centermarks, Centerlines, and Bisectors. This functionality is available only on the Inventor 2010 host.

2.8.2 Dimensions: Adoption enhancements

- For the Inventor 2010 host: Intent 2010 supports adoption of dimensions attached to Centermarks, Centerlines and Bisectors
- For all dimension types – support for Dimension styles and layers
- Additional parameters for dimensions (these appear in the Options category):

ArrowHeadsInside?

CenterText? (Linear and Angular)

OppositeAngle? (Angular)

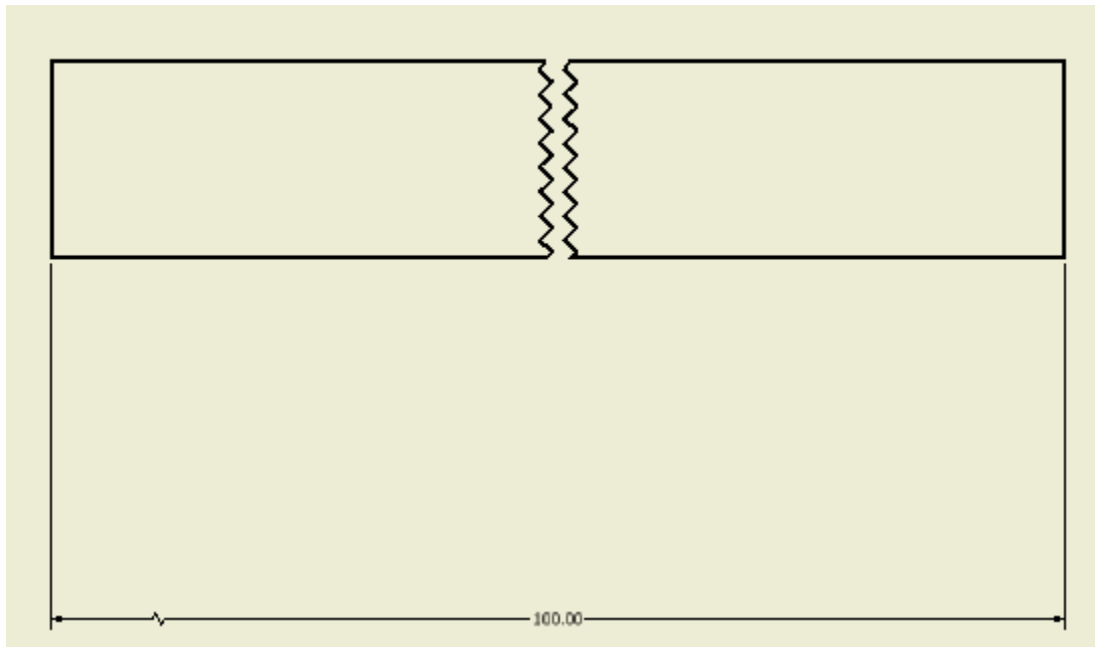
UseQuadrant? (Angular)

LeaderFromCenter? (Diameter, Radius)

SingleDimensionLine? (Diameter)

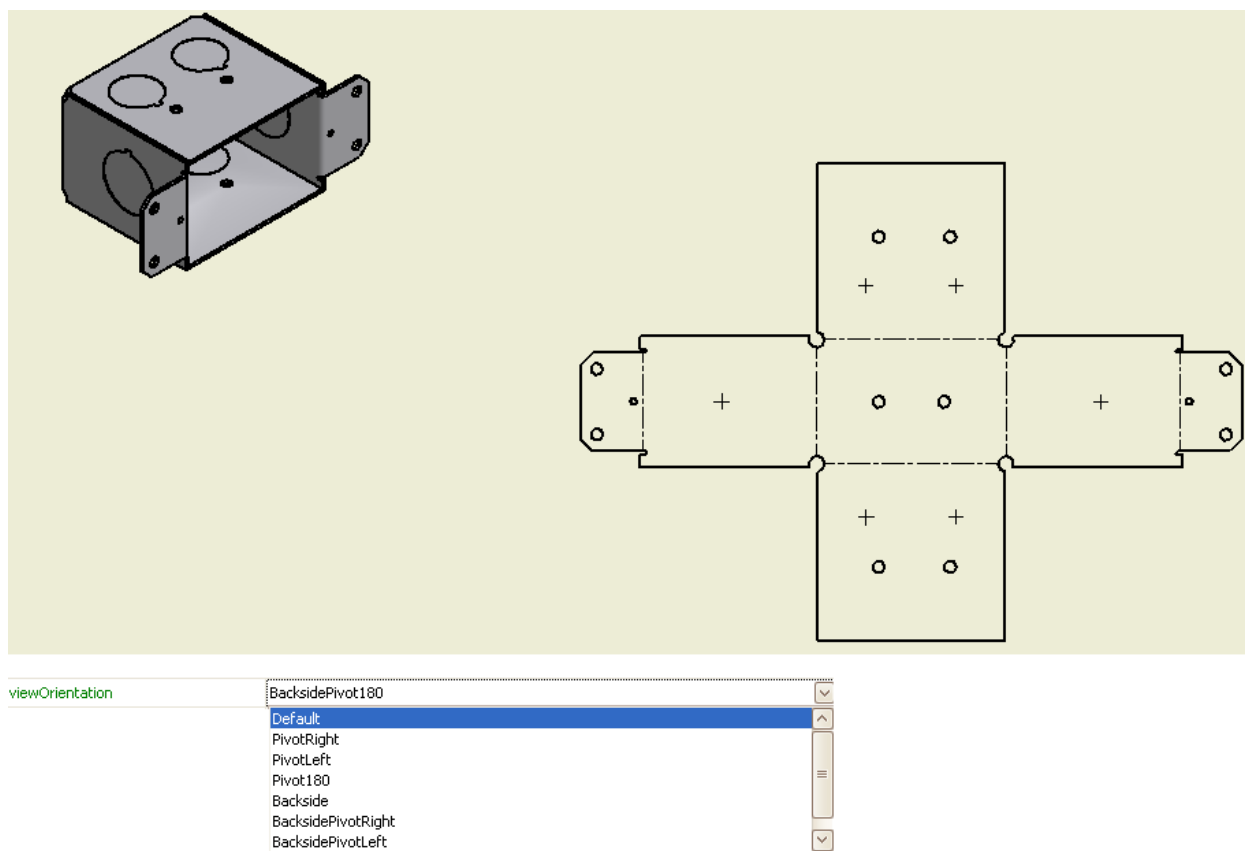
2.8.3 “Broken” view support

Available only on the Inventor 2010 host.



2.8.4 Sheet Metal Flat Pattern View

Sheet Metal Flat Pattern Views are supported via the **IvFlatPatternView** Design, which behaves very similarly to the **IvBaseView** Design (with exception of a different set of viewOrientation choices).



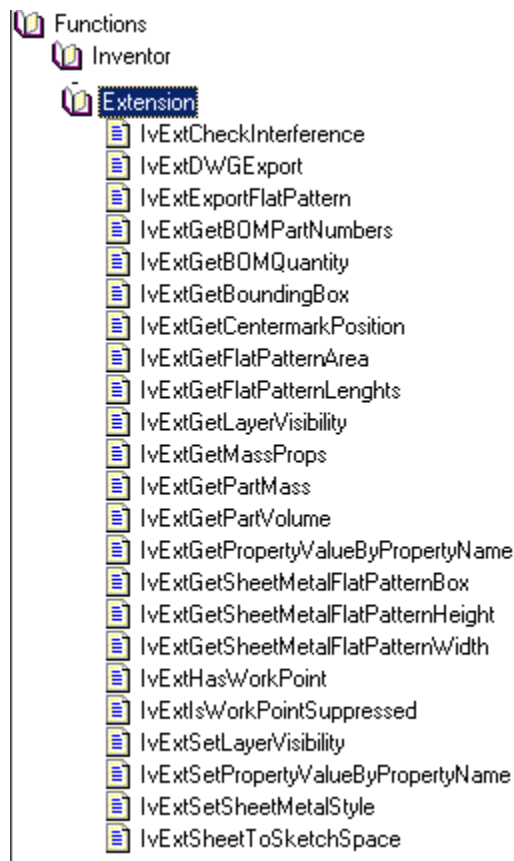
2.8.5 Hole Features

Hole Features are supported through the **IvAssemblyHoleFeature** and **IvPartHoleFeature** Designs.

2.8.6 Inventor Extension Utilities

Several useful extensions to the Inventor library, written by Jan Liska of Autodesk Consulting, have been incorporated into the product.

Detailed description of functions is available in the help file under Functions – Inventor – Extension.



Note: These functions have been changed slightly in behavior. Where applicable, the functions now return values in the units of measurement specified by the Intent project (controllable by the Intent Project Editor). Existing usage of these functions may need to be reviewed before use in Intent 2010.

2.8.7 Access to the part model during instance generation

Intent 2010 supports a long-desired ability to set a string into Part or Assembly files during the process of creation that allows them to contain information not available before the component is created, and which cannot be used after the component is finalized.

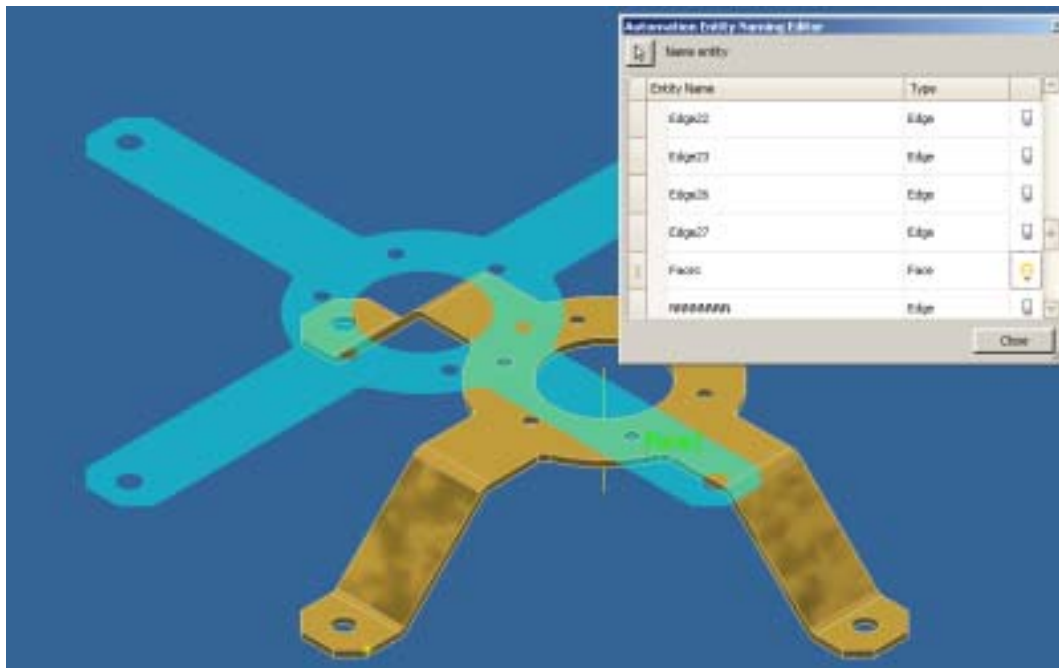
The most common usage of this is to set the Part Number based on parameters of the component, and for this reason, the ability uses the Part Number model for its description. However, any data that has the same requirements can be used.

For details see the documentation on `GetNewPartNumber`, and the following functions:

- `Iv_documentPropertyGet(docHandle as String) As Any`
- `Iv_DocumentPropertyPut(docHandle As String, propertyName as String, Value as Any,...)`
- `Iv_DocumentParamValue(dochandle as String, ParamName,...)`
- `Iv_DocumentParamValueAsString(dochandle as String, ParamName,...)`
- `Iv_PartNumberInLocale()`

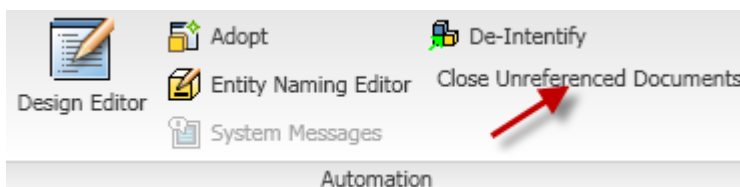
2.8.8 Entity Naming Editor can be used for Flat Patterns

As of Update 1, the Entity Naming Editor now supports naming items in the Flat Pattern. Note that the Editor combines all names for both flat and folded environments, so the same name cannot be used in both. In addition, it is not possible to tell in the Editor which environment a name belongs to. Furthermore, it is possible to highlight entities in one environment while looking at another. See the image below.



2.8.9 Close Unreferenced Documents

As of Update 2, there is a new command in the Inventor user interface, as well as a new API call (which currently only applies to Inventor). Invocation of this command will close all unreferenced documents which are open in Inventor. This can significantly reduce Inventor's memory footprint in certain situations. This happens automatically when an Intent Model is closed, but can be invoked at any time.



The API call is as follows:

```
HostAPI.Instance.CloseAllUnreferencedDocs();
```

2.8.10 DumpIKM

As of Update 2, there is a new command-line utility, `dumpIKM.exe`, which writes the Intent model contained in an Inventor IAM file to the standard output. This can be used to save dynamic changes to model independently of the IAM.

Usage:

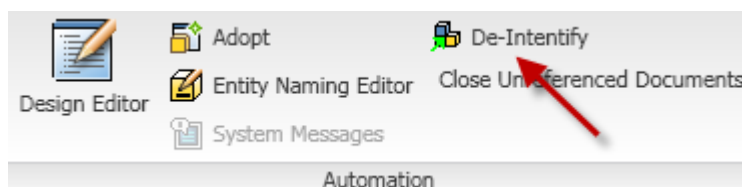
```
DumpIKM.exe filename.iam > filename.ikp
```

A word about the naming: “IKM” is the name for a generic Intent model file. “IKP” is the name for an Intent “project” file, now largely obsolete. The distinction is that an IKP always represents the top-level of the Intent model, and traditionally contained units and search path information. In Inventor and other hosts, this additional information is now handled separately (by the project file concept in Inventor), and is not used even if present.

DumpIKM sets `ERRORLEVEL` 0 on success, and 1 on failure (with an error message on the command line).

2.8.11 De-Identify

Sometimes it is required to send a set of Intent-generated Inventor files to other parties. If the receiving system does not have Intent loaded, certain messages may appear. In addition, it may be desirable to ensure that no proprietary knowledge is contained in these files. As of Update 2, a new De-Identify command is available:



This removes all Intent-related information from the Inventor files.

This capability is also available via the HostAPI:

```
HostAPI.Instance.DeIdentify(string destinationFolder, bool
overwriteExistingFiles);
```

2.8.12 Inventor API access from the Intent language

As of Update 2, the Inventor API can be called directly from the Intent language. For example:

```
Rule host As Any = Autodesk.Intent.IntentAPI.Instance.HostAPI
```

```
Rule iv As Any = host.HostApplication
```

```
Uncached Rule ivc As String = iv.Caption
```

```
Method Distance() As Number
```

```
    Dim d As Number = 0.0
```

```
    Dim occ1 = host.HostObjectFromRefChain("Block")
```

```
Dim occ2 = host.HostObjectFromRefChain(Cone.Refchain)
d = iv.MeasureTools.GetMinimumDistance( occ1, occ2)
```

```
Return d
```

```
End Method
```

```
' Created automatically via Convert to Design Rule
```

```
Child Block As :IvBlock
```

```
End Child
```

```
' Created automatically via Convert to Design Rule
```

```
Child Cone As :IvCone
```

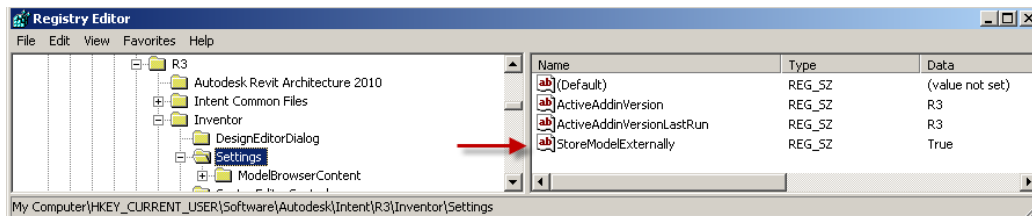
```
End Child
```

2.8.13 External Storage Mode

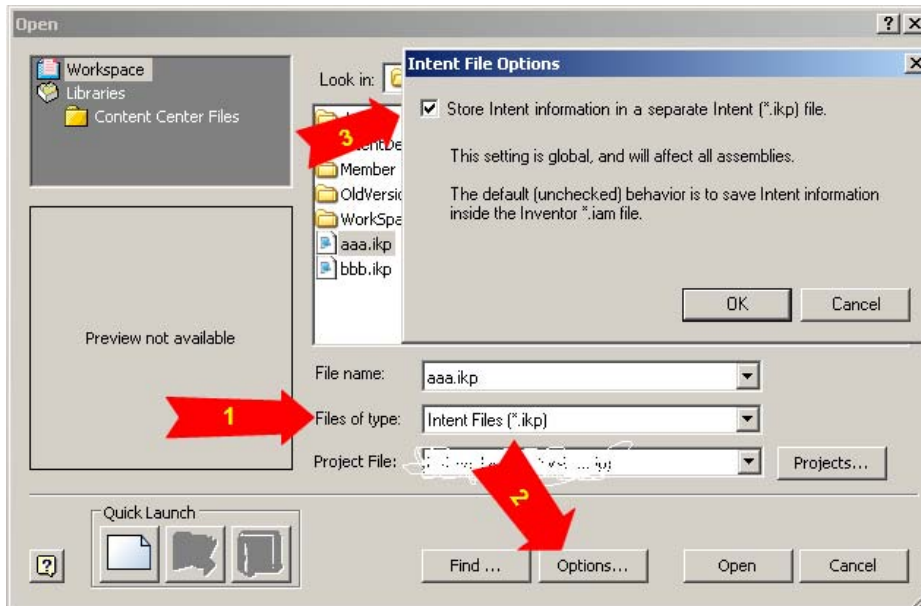
As of Update 2, it is now possible to store Intent model information externally; that is, not in Inventor IAM assembly files. This allows independent manipulation and management of Intent model information. This feature is implemented as a mode switch; the system is either storing Intent model information in the IAM files, or it is not. It is not possible to control this capability on a per-file basis.

The default mode is internal.

The mode is set in the system registry:



There is also a file open option that will set the mode. In order to apply the setting, the “type” of files being opened must be set to “Intent Files (*.ikp)”, then clicking on the options tab will bring up the option setting panel.

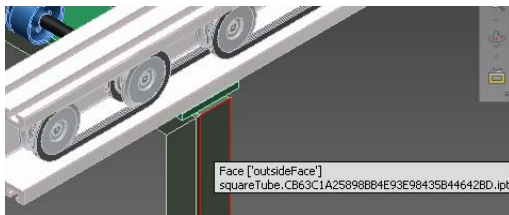


The mode setting is also controllable from the HostAPI. There is a Boolean property `StoreModelExternally`, which can be used to control this setting.

Note: although this is a global setting, as of Update 2 it is ignored by other hosts.

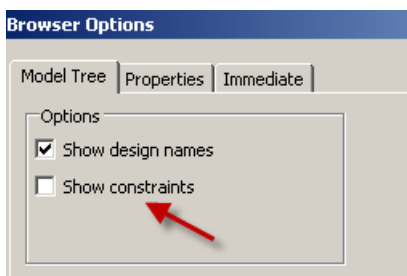
2.8.14 Entity Naming Editor Tooltips

In the Entity Naming Editor's selection mode, as of Update 4, a tooltip now shows the entity type, the Intent name, and the file name where the definition is stored.



2.8.15 New "Show constraints" ability

As of Update 4, it is now possible to toggle the visibility of constraints in the Model tree:

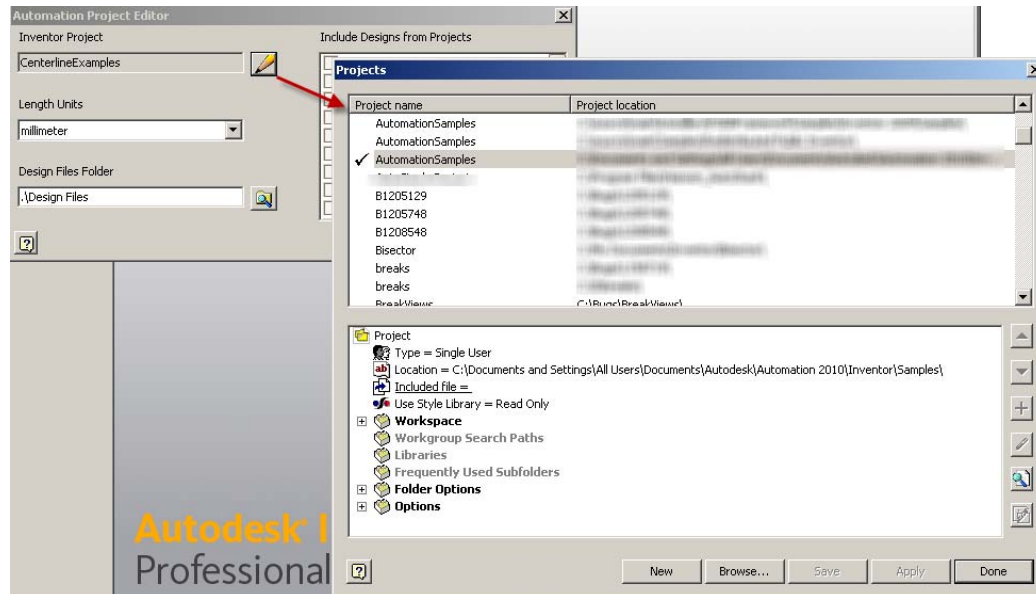


2.8.16 New Drawing IgnorePosition? Parameter

As of Update 4, Drawing views now support a parameter IgnorePosition?, which if true will force the view to keep a manual override of its position. The default value is false.

2.8.17 Project Activation from Intent Project Editor

As of Update 4, it is now possible to activate an Inventor project from the Intent Project Editor user interface. If the Inventor active project changes, Intent's Project Editor is updated.



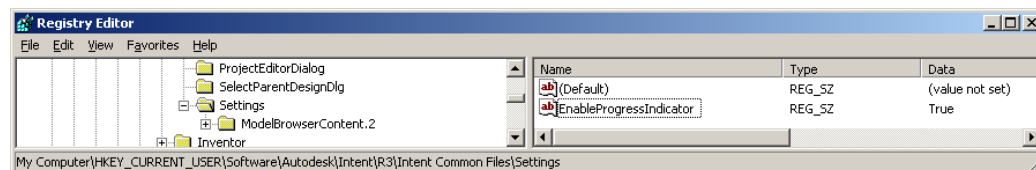
2.8.18 New Progress Indicator

As of Update 4, during a Model.Render operation, the status bar shows the current progress as shown:

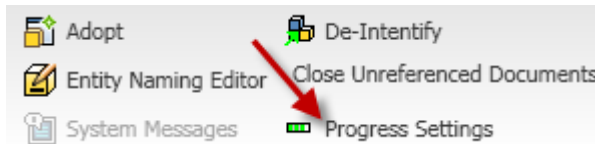
Processing Step 40 of 93 (43% done) Remaining Time: 10 sec

Intent does not pre-calculate the steps count, so the percentage may actually decrease as the model is rendered.

The progress indicator can be shown or not, via the user interface or programmatically. The setting itself is contained in the registry as shown:



The setting can be set from the Ribbon UI:



Or via the API:

```
IntentAPI.Instance.Options.IsRenderProgressEnabled
```

It is also possible to write your own progress indicator, by deriving from the abstract class Autodesk.Intent.ProgressIndicator.

[-] Syntax

C#

```
public abstract class ProgressIndicator
```

Visual Basic (Declaration)

```
Public MustInherit Class ProgressIndicator
```

The type exposes the following members.

[+] Protected Constructors

[-] Public Methods

	Name	Description
≡	OnEnd	Called when Indicator should be dismissed (if IntentAPI.Instance.Options.IsRenderPr
≡	OnProgress	Called when progress report is available (if IntentAPI.Instance.Options.IsRenderProg
≡	OnStart	Called when indicator starts (if IntentAPI.Instance.Options.IsRenderProgressEnabled

[-] Public Properties

	Name	Description
	StartTime	Time when indicator started. Useful for calculation of the progress

The simplest implementation is as follows:

Notes:

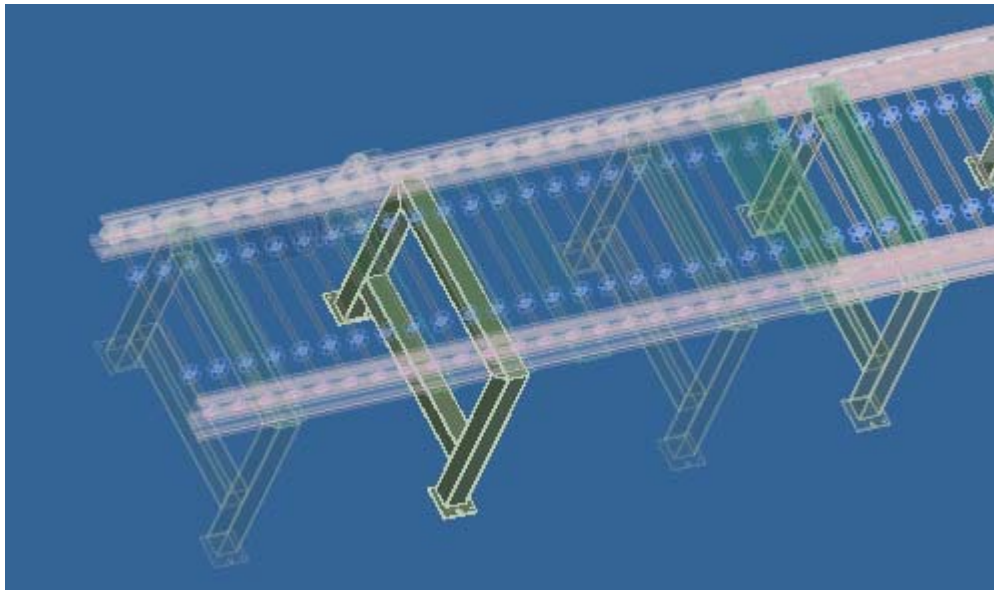
- Don't perform excessive calculations or graphics in the overridden functions, or Inventor performance will degrade.
- The OnProgress method is not called for every step, only when the percentage changes.
- Create the instance of your class after the IntentAPI has been initialized (check Autodesk.Intent.IntentAPI.Initialized).

In Intent language, the progress can be incremented based on actions during evaluation, as shown:

```
%%progressIncrementMaxRange( Length(renderedChildren) )  
For Each ch In renderedChildren  
    ch.renderSelf  
    %%progressIncrement()  
Next ch
```

2.8.19 New OverrideOpacity Parameter

As of Update 4, Intent now supports changing the opacity level on component occurrences. The valid range is 0.0 to 1.0 or NoValue. NoValue indicates “do not override”. The value 1.0 corresponds to opaque, and 0.0 is transparent. The following image shows several levels of opacity.



2.8.20 New Sheet Metal Support

As of Update 4, a new Design, IvSheetMetalPart, supports sheet metal parts in Inventor. It is derived from IvAdoptedPart, and adoption is the only supported way of bringing sheetmetal parts under the control of Intent. There is no support for creating sheetmetal features directly in Intent except as described below.

There is a separate document covering the new Sheetmetal capabilities that can be found in the Help directory, *SheetMetalSupport.pdf*.

The following table shows the new parameters supported by IvSheetMetalPart

ActiveSheetMetalStyle	Default
BaseFace	
HasFlatPattern?	<input checked="" type="checkbox"/>
Thickness	
UnfoldMethod	Default_KFactor

The parameters are described below.

Parameter	Adoptable?	Description
ActiveSheetMetalStyle	Yes	The name of the active sheet metal style. Default is NoValue, which indicates the style used in the source part should be maintained.
BaseFace	Yes	The face to utilize during flat pattern creation. Default is the empty string. Adoption may assign the actual name.
HasFlatPattern?	Yes	True if a flat pattern should exist. If false, any existing flat pattern will be deleted. Adoption will set the parameter according to whether the flat pattern exists in the source. [NOTE: Due to a defect in Inventor 2010, setting to false when the factory part contains a flat pattern will cause an error message to appear. See “Known Issues” for details.]
Thickness	No	Sets the thickness of the part. If set to NoValue, then the default thickness is used (which is part of the sheet metal style). In some situations, the thickness is a reference parameter in Inventor, and is not controllable by Intent.
UnfoldMethod	Yes	Name of the Unfold Method Style. The default is NoValue, which maintains the method found in the source file.

Several Uncached rules are supported:

FlatPatternBoundingBox	Returns the bounding box of the flat pattern in project units. Returns a list of two points, as {minPoint, maxPoint}.
FlatPatternWidth	Returns the width of the flat pattern in project units.
FlatPatternHeight	Returns the height of the flat pattern in project units.

The following Methods are defined:

FlatPatternLengths(useBottomFace? As Boolean, units As String)

Returns a list of outer and inner lengths of the flat pattern.

FlatPatternArea(useBottomFace? As Boolean, units As String)

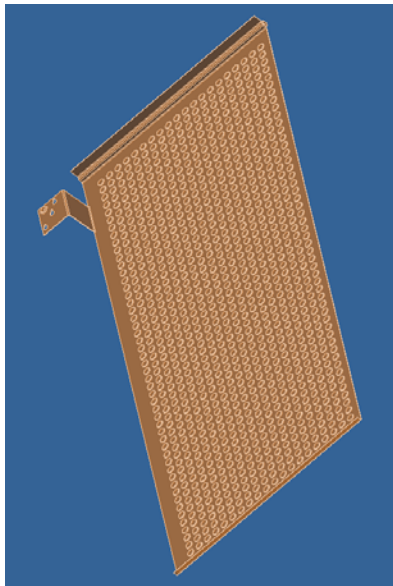
Returns the area of the flat pattern.

ExportFlatPattern(fileName As String, Optional formatStr As String = “FLAT PATTERN DXF?AcadVersion=R12”)

Writes the flat pattern to a file.

2.8.21 Punch Tool Feature for Sheet Metal

As of Update 4, a new capability to create punch features in sheetmetal parts is available.



A planar face or workplane is used to define a sketch plane, and its origin and orientation are defined by workpoint or vertex and workaxis or an edge, respectively. Then points on the sketch define where punch centers should be.

Punch Tool	
angle	0.0
face	PunchedFace
FeatureInputs	{{"Length","1.3 in"}, {"Hole_Diameter",0.9}, {"Slot_Width","0.3875 in"}, {"Fillet", "0.125 in"}}
iFeatureFile	iv_getPunchesPath() + "keyhole.ide"
Name	PunchBed
origin	Originpoint
Points	{ point(5,5,0),point(2,1,0),point(3,3,0)}
reverseXdirection?	<input type="checkbox"/>
xAxis	AnXAxis

The parameters are described in the following table.

Name	The Name of the punch tool feature.
Face	The name of a planar face or work plane, on which the sketch will be

	created.
Origin	The name of a vertex or a work point, which defines the sketch origin.
xAxis	The name of an edge or a work axis, which defines the x-direction in the plane.
ReverseXDirection?	If true, reverse the direction of the xAxis.
Points	The list of points that specify the punch centers.
iFeatureFile	The file name which contains the punch definition.
Angle	The angle of rotation of the punch relative to the x-axis.
iFeatureInputs	<p>A list of feature inputs. The list is formatted as { {stringInputName1, value1}, {stringInputName2, value2}, ... } The name is the name of the parameter in the iFeature file, the value can be numeric or a string, where the string is interpreted as an Intent expression.</p> <p>For table-driven punches, the active row may be specified as {rowNumber}.</p> <p>The default is the empty list.</p>

2.9 ODBC Support in 64bit Installations

As of Update 1, the ODBC library is now supported in 64bit installations, with the necessary restriction that the DAO technology is not supported (because it does not exist for 64bit).

2.10 ADO.Net Support

As of Update 1 Intent now supports ADO.Net, the newer form of database access from Microsoft. There is a separate document describing this new feature, see the document in C:\Program Files\Autodesk\Inventor Automation Professional 2010\Library\Standard\adonetlib.

ADONetLib is supported on all platforms, and is independent of CAD hosts.

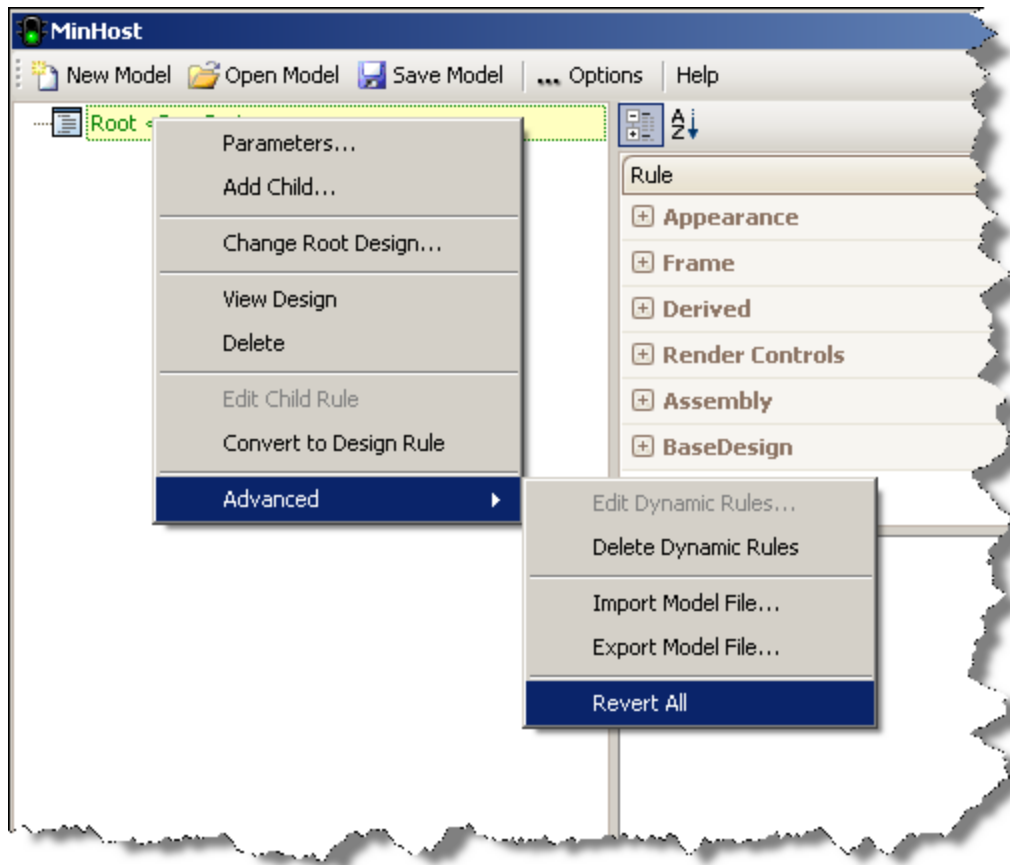
2.11 HostObjectFromRefchain method

As of Update 2, a new HostAPI method is available to obtain the host entity object for a given Intent object. This simplifies access to Host models from within the Intent language when using direct .Net referencing capabilities. For example:

```
Rule host As Any = Autodesk.Intent.IntentAPI.Instance.HostAPI
Dim occ1 = host.HostObjectFromRefChain("Block")
Dim occ2 = host.HostObjectFromRefChain(Cone.Refchain)
```

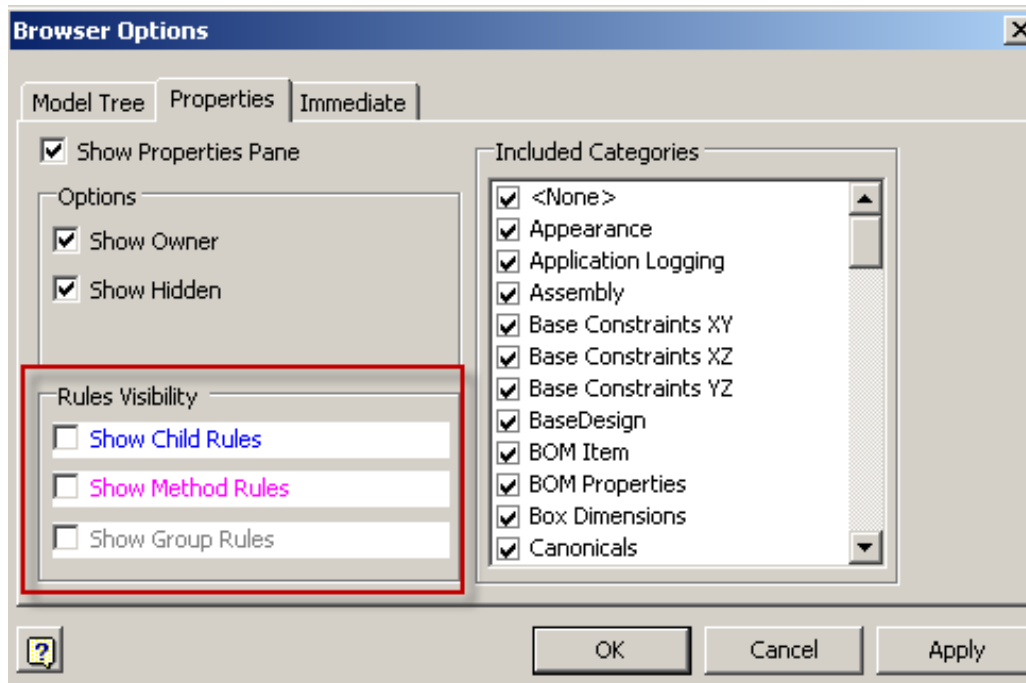
2.12 Revert All Designs

As of Update 2, there is a new global command available on the right-click context advanced menu for Parts in the assembly tree. Although this is available on all nodes in the tree, its effect is global. All currently loaded designs will be re-loaded from source. This is convenient for users making mass changes to designs in external editors, or in situations where the Design Editor is not available or in use.





2.13 Visibility of Child, Method, and Group Rules

As of Update 3, Child, Method, and Group Rules can now be made visible in the Properties window. This is controlled through the Options dialog box:



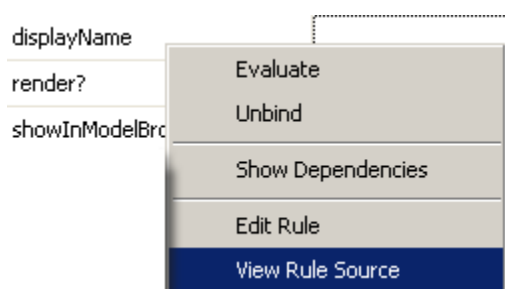
The colors are used to show the types of rules in the Properties window:

childListLength		%%BaseDesign
children	{Root.Window, Root.PGTDB, Roo... 	%%BaseDesign
commaSeparatedList		%%HostRenderMixin
ComponentSpec		%%IvAssemblyDocument
Configuration		XMLInputMixin
CustomDHSashLoc		XMLInputMixin
cyclicNext		%%BaseDesign
cyclicPrevious		%%BaseDesign
DefaultBOMStructure	:Normal	%%IvBOMInfoMixin
descendantOf?		%%HostAssemblyMixin
designName		%%BaseDesign
displayName	"Root" 	%%BaseDesign
DPRating		XMLInputMixin
EgressRequired		XMLInputMixin
FindInList		XMLInputMixin
First		%%BaseDesign

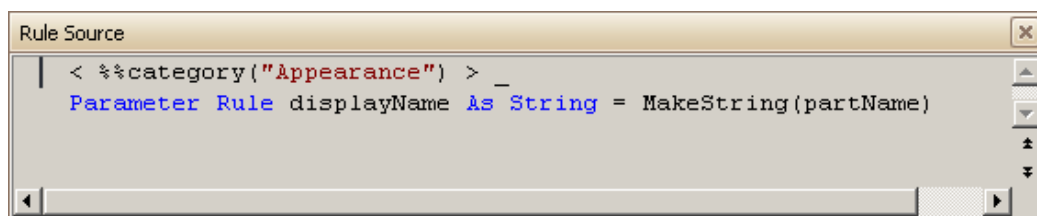
2.14 Rule Source Window

As of Update 3, you can now pop up a read-only rules source window that will track the selection of a rule in the Properties window and will show the source code for that rule.

Right-clicking on the Rule in the Properties Pane shows a new menu item:



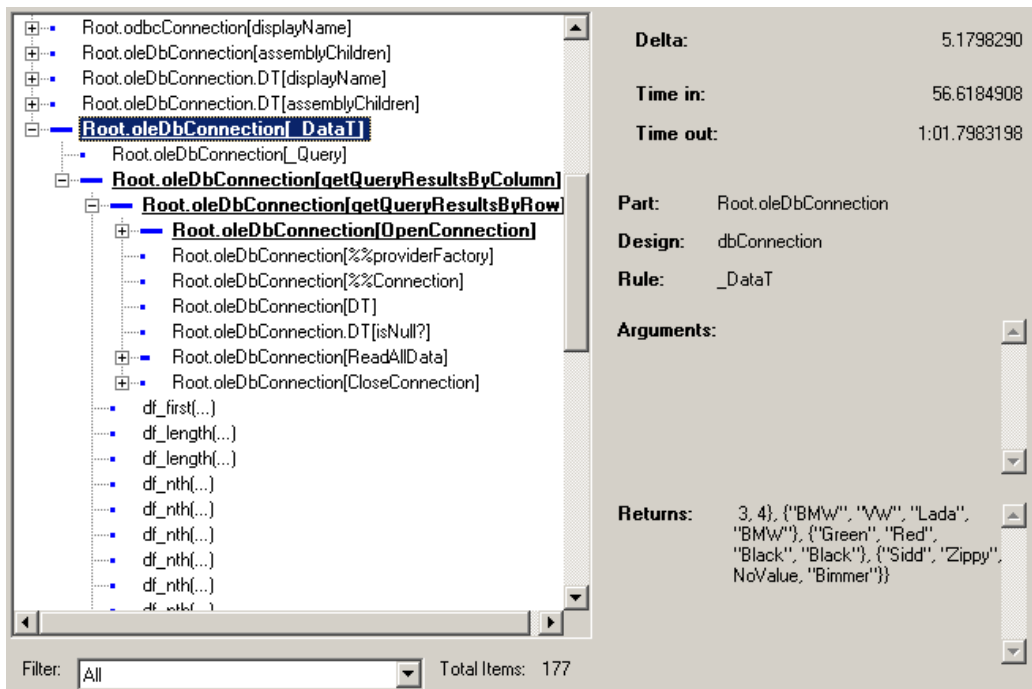
When selected, the floating read-only Rule Source pane appears:



When you click on other Rules, the content of the control is updated.

2.15 Intent Tracer

As of Update 3, there is a new utility for profiling your Intent models to see where time is being spent. It is called the Intent Tracer:



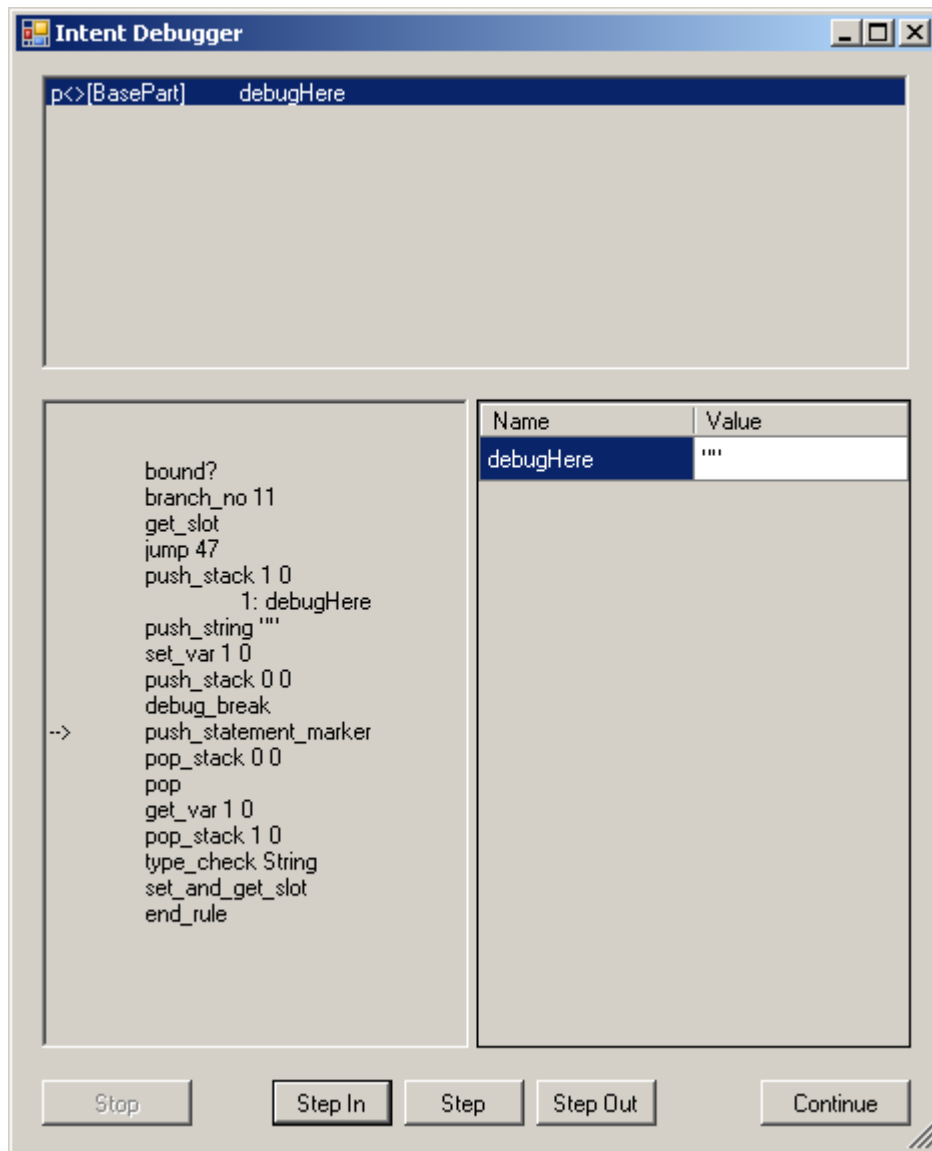
There is documentation for it in the installed Help directory in the file IntentTracer.pdf.

2.16 New Assembly (dll) for Reference in .Net projects

As of Update 3, Autodesk.Intent.Exceptions.dll is no longer produced. Everything that was in that dll has been moved into Autodesk.IntentBase.dll, and that should be referenced instead. Also, the definitions for Point, Vector, and Frame were moved to this new dll.

2.17 New Intent Run-Time Debugger

As of Update 3, there is a new run-time debugger integrated into the development environment:



There is documentation for it in the installed Help directory in the file IntentDebugger.pdf.

2.18 UI Performance Improvements

As of Update 3, there are significant improvements in the performance of the Assembly Tree and Part Editor UI components.

2.19 New Deployment Installation Option

As of Update 4, deployment installations are now supported. A deployment installation is a pre-created installation created by a system administrator that can be placed on a server, from where client installations are made, which automatically include settings set by the administrator when

the deployment is built. This is a standard way of distributing network seats or multi-seat standalone seats.

Deployment installation is invoked by the Create Deployments option on the installation launch window. Details of deployment are beyond the scope of this document, but if Intent is to be deployed in a corporate setting where the host (Inventor or AutoCAD) is also distributed via deployment, the process is identical.

2.20 New Features in AutoCAD

2.20.1 New application sample projects

As of Update 2, two sample projects are included in the distribution for AutoCAD developers.

Simple is the simplest possible “Hello World” type of application to ensure that your development environment, AutoCAD, and the basic invoking and calling of Intent inside AutoCAD is working.

SimpleUI is a simple full application, with the ability to create a new project, save and load changes, with a custom user interface using the AutoCAD “Palette” object to control it.

In the Windows XP environment, the samples are located in

C:\Documents and Settings\All Users\Documents\Autodesk\Automation 2010\AutoCAD

In the Windows Vista environment, the samples are located in

C:\Users\Public\Documents\Autodesk\Automation 2010\AutoCAD

2.21 Changes to Return statement behavior

As of Update 4, the behavior of the Return statement in the Intent language has been corrected to match its original design. Prior to Update 4, Return could not be used inside a Begin/End expression to return the value of the expression; instead, it would return from the rule, method or function containing it (a “global” return). This effectively reduced the utility of the Begin/End construct, and created situations where the IntentUp translator created source that did not work properly.

With the change, a Return inside a Begin/End block will always be a local return, returning the value of the expression. It is not possible to have a global return inside a Begin/End block. This change also allows the commonly-desired Begin/End blocks as values for Child Rule parameters, without getting the “no target for Return statement” error.

Unfortunately, a defect was introduced in Update 4 because of this change, which was fixed in Update 5. Update 4 can cause data stack corruption, and should not be used in production.

3 Changes

The following fixes over prior versions are included in Intent 2010. This listing includes only higher-severity issues that were addressed (primarily) by code-fixes.

Did	Summary	Se...	Generic1	Generic2	Feature	Status	SubStatus	Created	Week Created
1145398	64BIT: ATAP CRASHED WHEN RUN MAKEIKB ON 64 BIT MACHINE.	1	.	.	Core: MakeIKB	Closed	CODE FIX	10/24/2008	10/19/2008
1192079	AUTOCAD + INTENT DOESNT WORK	1	.	.	Acad: General	Closed	CODE FIX	1/20/2009	1/18/2009
1200165	AUTOMATION FAILS TO OPEN ASSEMBLY WITH TRANSACTION ERROR	1	.	.	Inv: Assembly	Closed	CODE FIX	2/12/2009	2/8/2009
1203221	0000272: CRASH ON DOUBLE CLICKING THE ADOPT BUTTON ON RIBBON FOR ALREADY ...	1	.	.	Inv: Adopt	Closed	CODE FIX	2/23/2009	2/22/2009
1204420	0000280: CRASHES WHEN INVALID CHARACTERS USED IN DESIGN NAME OF 'NEW AUT...	1	.	.	Inv: UI	Closed	CODE FIX	2/26/2009	2/22/2009
1205134	0000300: CRASH WHEN SINGLE QUOTES ARE PRESENT IN THE DESIGN NAME	1	.	.	Inv: Adopt	Closed	CODE FIX	3/2/2009	3/1/2009
1206800	0000305: RADIAL DIMENSION NOT ADOPTED (DIMENSION ADOPTION)	1	.	.	Inv: Adopt	Closed	CODE FIX	3/9/2009	3/8/2009
1206815	0000314: CENTER LINE ADOPTION PROBLEM (UNABLE TO ADOPT THE CENTER LINE CRE...	1	.	.	Inv: Adopt	Closed	CODE FIX	3/9/2009	3/8/2009
1207128	0000329: ERROR WHILE CREATING BREAK VIEW	1	.	.	Inv: Drawings	Closed	CODE FIX	3/10/2009	3/8/2009
1207130	0000331: ERROR WHILE CREATING BREAK VIEW: BREAKVIEW IS CREATED EVEN IF THE ...	1	.	.	Inv: Drawings	Closed	CODE FIX	3/10/2009	3/8/2009
1207131	0000332: ERROR WHILE CREATING BREAK VIEW: BREAKVIEW IS NOT CREATED BY GIV...	1	.	.	Inv: Drawings	Closed	CODE FIX	3/10/2009	3/8/2009
1207132	0000333: ERROR WHILE CREATING BREAK VIEW	1	.	.	Inv: Drawings	Closed	CODE FIX	3/10/2009	3/8/2009
1207137	0000340: CENTER LINE NOT ADOPTED (DIMENSION ADOPTION)	1	.	.	Inv: Adopt	Closed	CODE FIX	3/10/2009	3/8/2009
1207445	0000349: UNABLE TO ADOPT THE CONSTRAINT IN THE SUB ASSEMBLY	1	.	.	Inv: Adopt	Closed	CODE FIX	3/11/2009	3/8/2009
1207446	0000351: APPLICATION HANG: IF DRAWING IS GENERATED AND ACTIVE THEN IF DRAW...	1	.	.	Inv: Assembly	Closed	CODE FIX	3/11/2009	3/8/2009
1207455	0000361: UPDATING THE SECTION VIEW CRASHES THE INVENTOR	1	.	.	Inv: Drawings	Closed	CODE FIX	3/11/2009	3/8/2009
1207456	0000363: APPLICATION ERRORS ON ENTERING 0 INTO THE TEXT FONT SIZE IN BROWS...	1	.	.	Core: UI	Closed	CODE FIX	3/11/2009	3/8/2009
1207476	UNARY MINUS LOSES PARENS IN CONVERSION	1	.	.	Core: IntentUp	Closed	CODE FIX	3/11/2009	3/8/2009
1208550	0000382: NOT ABLE TO GET THE OBJECT OF INTENT □IHOSTAPI□ IN VB.NET 2008	1	.	.	Core: API	Closed	CODE FIX	3/16/2009	3/15/2009
1209962	ACAD CRASHES DURING ACAD_MAKECYLINDER	1	.	.	Acad: Libraries	Closed	CONFIGURATION	3/23/2009	3/22/2009
1210517	VOL_COPY ERROR WHEN EVAL UNKNOWN RULE NAME	1	.	.	Core: Immediate Window	Closed	CODE FIX	3/25/2009	3/22/2009
1211062	EVALUATION OF .NET CALL ALWAYS CRASHES MINHOST	1	.	.	MinHost: General	Closed	CODE FIX	3/27/2009	3/22/2009
1211209	PUT CORRECT PART NUMBER IN RAR FILES.	1	.	.	Core: Installation & Licensing	Closed	CODE FIX	3/30/2009	3/29/2009
1076681	INVENTOR HAD NO RESPONSES WHEN COMPILE AND SAVE DESIGN AFTER DELETING SO...	2	.	.	Inv: Drawings	Closed	CODE FIX	5/8/2008	5/4/2008
1095146	REGRESS: CANT CHANGE ASSY-HOLE FROM :AUTOSELECT TO THROUGHALL	2	.	.	Inv: Features	Closed	CODE FIX	7/13/2008	7/13/2008
1098096	AUTOMATION INITIALIZATION ERROR MESSAGE POPPED UP WHEN NEW PROJECT	2	.	.	Inv: General	Closed	NOT REPRODUCIBLE	7/21/2008	7/20/2008
1102129	64-BIT: THE ASSEMBLY WITH W/ASSEMBLYHOLEFEATURE COULD NOT BE OPEN IF USE T...	2	STOPBETA	BB1	Inv: Features	Closed	NOT REPRODUCIBLE	7/30/2008	7/27/2008
1102542	CRASH WHEN ADOPTING A DIMENSION INVOLVING GEOMETRY FROM ASSEMBLY FEATURE	2	.	BB1	Inv: Features	Closed	CODE FIX	7/31/2008	7/27/2008
1102544	NO WAY TO EDIT BROKEN HOLE FEATURES (AKA NO CONTEXT MENU ON ERRORED HOL...	2	.	BB1	Inv: Features	Closed	NOT REPRODUCIBLE	7/31/2008	7/27/2008
1107515	64-BIT: AUTOMATION CRASHED WHEN RUN MAKEIKB ON 64 BIT VISTA	2	.	.	Core: MakeIKB	Closed	CODE FIX	8/13/2008	8/10/2008
1113109	INVENTOR CRASHES WHEN ADDING A VERY SMALL SPHERE(32 BIT ONLY)	2	.	.	Core: Error Dialog / Debugger	Closed	CODE FIX	8/26/2008	8/24/2008
1116500	CHILD ADDED TO MODEL TREE ON ERROR	2	.	.	Core: Part Editor	Closed	CODE FIX LINKED	9/3/2008	8/31/2008
1116516	WHEN OPENING AN ASSEMBLY WHICH REFERENCES AN INVALID DESIGN YOU GET CANN...	2	.	.	Core: Error Dialog / Debugger	Closed	CODE FIX	9/3/2008	8/31/2008
1118806	BOM IS NOT RIGHT AFTER DELETING PARTS LIST	2	CCB APPRO...	.	Inv: Drawings	Closed	CODE FIX	9/8/2008	9/7/2008
1120967	REGRESS:CLOSE MODEL AFTER INCREMENTAL ADOPT IMATECONSTRAINT RESULT CRA...	2	CCB PENDING	CCB	Inv: Adopt	Closed	NOT REPRODUCIBLE	9/11/2008	9/7/2008
1121111	64BIT: AUTOMATION CRASHED WHEN RUN INTENTUP ON 64 BIT VISTA OS.	2	README	.	Core: IntentUp	Closed	CODE FIX	9/12/2008	9/7/2008
1122802	UNABLE TO CREATE DRAWING WITH DRAWINGSAMPLE ON INV2010	2	.	.	Inv: Drawings	Closed	NOT REPRODUCIBLE	9/16/2008	9/14/2008
1123359	REGRESS:SYSTEM DESIGN MISSING FROM DE AND UNCLASSIFIED CATEGORY APPEAR...	2	.	.	Core: Design Editor	Closed	CODE FIX	9/17/2008	9/14/2008
1124974	INVENTOR 2010 COULD NOT LOAD THE AUTOMATION ADDIN	2	.	.	Core: Installation & Licensing	Closed	CODE FIX	9/21/2008	9/21/2008
1124974	INVENTOR 2010 COULD NOT LOAD THE AUTOMATION ADDIN	2	.	.	Core: Installation & Licensing	Closed	CODE FIX	9/21/2008	9/21/2008
1126729	REGRESS:SAVING PATTERNSAMPLE BRINGS INVENTOR ERROR(INV2010)	2	.	.	Inv: Assembly	Closed	AT VENDOR	9/24/2008	9/21/2008
1131910	CANNOT GET 'MASTER'	2	.	.	Inv: Assembly	Closed	CODE FIX	10/6/2008	10/5/2008
1132159	REGRESS:CONVERT TO DESIGN RULE DOESNT WORK.	2	.	.	Core: UI	Closed	CODE FIX	10/6/2008	10/5/2008
1132364	REGRESS:UNHANDLED EXCEPTION WHEN TRY TO OVERRIDE AN EXISTING IAM.	2	.	.	Inv: Assembly	Closed	CODE FIX	10/7/2008	10/5/2008
1133235	UNHANDLED EXCEPTION POPPED UP WHEN ADD THE PART HOLE FEATURE MANUALLY	2	.	.	Inv: Features	Closed	CODE FIX	10/8/2008	10/5/2008
1135789	NO ENTITY NAME OF IPARTHOLEFEATURE EXISTS IN THE ENE	2	.	.	Inv: Features	Closed	CODE FIX	10/12/2008	10/12/2008
1137252	REGRESS:GENERALNOTE COULD NOT BE ADDED USING THE USER DEFINED TEXT STYLE.	2	.	.	Inv: Drawings	Closed	CODE FIX	10/14/2008	10/12/2008
1138234	REGRESS:UNHANDLED EXCEPTION POPPED WHEN RUN INTENTUP	2	.	.	Core: Utilities	Closed	CODE FIX	10/15/2008	10/12/2008
1141391	REGRESS: SEVERAL DESIGNS CANNOT BE INSERTED WITH VARIOUS EXCEPTION.	2	.	.	Acad: Libraries	Closed	CODE FIX	10/19/2008	10/19/2008
1142399	REGRESS:CANT ADD DYNAMIC PART/ASM FEATURES CHILDREN.	2	.	.	Inv: Features	Closed	CODE FIX	10/20/2008	10/19/2008
1142429	UNHANDLED EXCEPTION WHEN SWITCH BETWEEN CUSTOM PE AND STANDARD PE	2	.	.	Core: Part Editor	Closed	CODE FIX	10/20/2008	10/19/2008
1146870	REGRESS:BUG ALERT "ONE OR MORE FACE LEAKED" WHEN OPEN MODEL THAT CONTAI...	2	.	.	Inv: Features	Closed	NOT REPRODUCIBLE	10/27/2008	10/26/2008
1148230	TOO LONG STRING IN THE HEADER COMMENT MADE MAKEIKB CRASH.	2	.	.	Core: MakeIKB	Closed	CODE FIX	10/29/2008	10/26/2008
1148971	UNHANDLED EXCEPTION IS THROWN WHEN SELECT FIND IN WINDOW ON A DELETED N...	2	.	.	Inv: General	Closed	CODE FIX	10/29/2008	10/26/2008
1149044	THROW ERROR DIALOG WHEN TRYING TO OPEN SECOND AUTOMATION FILE	2	.	.	Inv: General	Closed	CODE FIX	10/29/2008	10/26/2008
1154756	CANNOT ADD EXTRUDE FEATURE WHEN NESTED PROFILE IS PRESENT	2	.	.	Inv: Features	Closed	CODE FIX	11/7/2008	11/2/2008
1158719	INVENTOR CRASHES AFTER SUSPENDING AND TURNING ON UPDATE OPERATIONS, WHE...	2	.	.	Inv: Drawings	Closed	CODE FIX	11/12/2008	11/9/2008
1158733	ERROR MESSAGE IS WRONGS.	2	.	.	Core: Error Dialog / Debugger	Closed	CODE FIX	11/12/2008	11/9/2008
1162528	REGRESS: SAVE DESIGN DIALOG SHOULDN'T POP UP AFTER SAVE FROM DE.	2	.	.	Inv: General	Closed	CODE FIX	11/18/2008	11/16/2008
1164221	ADOPT RUNS AND RUNS AND RUNS OUT OF MEMORY	2	.	.	Inv: Adopt	Closed	NOT REPRODUCIBLE	11/20/2008	11/16/2008
1165600	INVENTOR CRASHED WHEN COMMENT/JMCOMMENT THE CHILD DEFINITION FOR TWIC...	2	REINST	.	Core: Error Dialog / Debugger	Closed	CODE FIX	11/23/2008	11/23/2008
1205127	0000281: "CHARACTER IN PART PARAMETER'S COMMENTS CREATES PROBLEM IN ADOP...	2	.	.	Inv: Adopt	Closed	CODE FIX	3/2/2009	3/1/2009
1205129	0000282: PROBLEM WHILE ADOPTING DERIVED PART WITH SUPPRESSED LINK	2	.	.	Inv: Adopt	Closed	CODE FIX	3/2/2009	3/1/2009
1205138	0000298: OFFICE SAMPLE NOT ABLE TO GENERATE EXCEL OR WORD DOCUMENT	2	.	.	Core: Office Library	Closed	CODE FIX	3/2/2009	3/1/2009
1205142	0000299: DRAWING GENERATION PROBLEMS IN STOOL SAMPLE	2	REINST	.	Core: Help and Docs	Closed	CODE FIX	3/2/2009	3/1/2009
1206823	0000324: WORDDOCUMENT OVERWRITES EVEN IF SET TO TRUE OR FALSE	2	.	.	Core: Office Library	Closed	CODE FIX	3/9/2009	3/8/2009
1207444	0000348: ERROR WHILE BALLOON GENERATION: UNABLE TO CREATE BALLOON BY SELE...	2	.	.	Inv: Drawings	Closed	CODE FIX	3/11/2009	3/8/2009
1208553	0000384: BALLOON POSITION IS NOT RELEVANT WITH THE SUPPLIED PARAMETERS	2	.	.	Inv: Drawings	Closed	CODE FIX	3/16/2009	3/15/2009
1208554	0000386: LEADERNOTE: LAYERS NOT ADDED TO THE LEADER NOTE	2	.	.	Inv: Drawings	Closed	CODE FIX	3/16/2009	3/15/2009
1208559	0000398: VISIBLE? PROPERTY: THE SUB ASSEMBLY DOES NOT HIDE	2	.	.	Inv: Drawings	Closed	CODE FIX	3/16/2009	3/15/2009
1208569	0000412: ISSUES_WITH THE ` CHARACTER	2	.	.	Core: Design Editor	Closed	CODE FIX	3/16/2009	3/15/2009
1210485	0000431: STOOL SAMPLE CRASHES THE INVENTOR	2	REINST	.	Inv: Assembly	Closed	CODE FIX	3/25/2009	3/22/2009
1211012	MINHOST IS NOT SUPPORTING RUNTIME MODE	2	.	.	MinHost: General	Closed	CODE FIX	3/27/2009	3/22/2009
784257	REPAIR INSTALL RECORDS INCORRECT INSTALLDIR LOCATION IF CUSTOM LOCATION ...	2	.	.	Core: Installation & Licensing	Closed	CODE FIX	5/1/2006	4/30/2006

3.1 The following high-severity defects were addressed in Update 1:

DID	Summary	Generic1	Generic2	Severity	Feature	Status	SubStatus	Created	Week Created	Week Cl...
1106799	CENTERLINES SOMETIMES DONT GO AWAY	STOPSHIP-2	.	2	Invr: Drawings	Fixed	Back to Tech	8/11/2008	8/10/2008	12/31/9999
1216208	MY BLOWER -- DRAWING IS NOT CLOSING	.	.	1	.	Will Not Fix	USER ERROR	4/22/2009	4/19/2009	5/3/2009
1209422	THE MODEL NEEDS TO BE UPDATED WHEN TRYING TO CREATE DRAWING	WATT 2.5,	2	Invr: Drawings	Will Not Fix	USER ERROR	3/19/2009	3/15/2009	5/3/2009
1216207	ENTITY NAMING PROBLEMS WITH THE FLAT PATTERNS	.	.	1	.	Closed	CODE FIX	4/22/2009	4/19/2009	4/26/2009
1216211	PROBLEMS WITH IASSEMBLYEXTRUDEFEATURE	.	.	1	.	Duplicate/Closed	Back to Tech	4/22/2009	4/19/2009	4/19/2009
1215774	HINT_STARTUP GETS REGISTRY ERROR	.	.	1	Acad: General	Closed	CONFIGURATION	4/21/2009	4/19/2009	4/19/2009
1212989	ENLIMS ARE NOT PROCESSED CORRECTLY IN .NET SYNTAX EXTENSION	.	.	1	Core: Kernel	Closed	CODE FIX	4/8/2009	4/5/2009	4/12/2009
1212408	ARRAY-TYPES ARE	.	.	1	Core: Kernel	Closed	CODE FIX	4/6/2009	4/5/2009	4/12/2009

3.2 The following high-severity defects were addressed in Update 2:

DID	Summary	Generic1	Generic2	Feature	Status	SubStatus	Created	Week Created	Week Closed	Tech
1212319	CANNOT ACCESS INVENTOR PROGRAMMING MODEL THROUGH .NET CALLS	.	.	Core: Kernel	Closed	CODE FIX	4/4/2009	3/29/2009	6/7/2009	Alex Korzun
1221106	WRONG ERROR-HANDLER IN INVENTOR	1	.	Invr: UI	Closed	CODE FIX	5/12/2009	5/10/2009	5/31/2009	Alex Korzun
1223620	MINHOST NEEDS TO ADD MODEL LIBRARY WHEN OPEN FILE	1	.	MinHost: General	Closed	CODE FIX	5/21/2009	5/17/2009	5/24/2009	Alex Korzun
1223660	IMMEDIATE WINDOW EVALUATIONS NEED TO CAUSE UI TO BE REFRESHED	1	.	MinHost: General	Closed	CODE FIX	5/21/2009	5/17/2009	5/24/2009	Alex Korzun
1226334	INTENTUP STRINGS CHOICE COMBINATION PROBLEM	1	.	.	Closed	CODE FIX	6/2/2009	5/31/2009	5/31/2009	Jack Gregory
1226396	EMPTY MIDON LIST STILL PRODUCING COLON IN INTENTUP	1	.	Core: IntentUp	Closed	CODE FIX	6/2/2009	5/31/2009	5/31/2009	Jack Gregory
1106799	CENTERLINES SOMETIMES DONT GO AWAY	2	STOPSHIP-2	Invr: Drawings	Closed	CODE FIX	8/11/2008	8/10/2008	5/10/2009	Alex Korzun

3.3 The following high-severity defects were addressed in Update 3:

DID	Summary	Status	Se...	Generic1	Generic2	Feature	SubStatus	Created	Week Created	Week Closed
1227665	NLA AND RELATED NOT INSTALLED BY INSTALLER	Closed	1	.	.	Core: Installation & Licensing	CODE FIX	6/8/2009	6/7/2009	8/16/2009
1233659	NOT INSTALLING ON AUTOCAD MECHANICAL 2010	Closed	1	.	.	Core: Installation & Licensing	CODE FIX	6/15/2009	6/14/2009	8/16/2009
1236331	MULTILINE NOT REALLY SUPPORTED	Closed	1	.	.	Core: Immediate Window	CODE FIX	6/24/2009	6/21/2009	8/16/2009
1236939	CONSTRAINTS DEFINED IN INTENT MODEL BUT MISSING IN INVENTOR ASSEMBLY AFTER...	Will Not Fix	1	.	.	Invr: Assembly	AS DESIGNED	6/26/2009	6/26/2009	8/9/2009
1246106	FOR STATEMENT DOES NOT AUTO-DECLARE ITS COUNTER	Closed	1	.	.	Core: Kernel	CODE FIX	8/11/2009	8/11/2009	8/11/2009
1248602	ERROR 2 RETURNED DURING STARTUP ON SWEDISH XP	Will Not Fix	1	.	.	Core: Installation & Licensing	.	8/20/2009	8/20/2009	8/20/2009
1248672	ALLOW OPTION TO TURN OFF TIMESTAMPS IN SYSTEM MESSAGES	Closed	1	.	.	Core: API	CODE FIX	8/20/2009	8/20/2009	8/20/2009
1249298	MINHOST SHOULD REMEMBER THE SEARCH PATH	Closed	1	.	.	MinHost: General	CODE FIX	8/24/2009	8/24/2009	8/24/2009
1257235	EXTRUDESOLID USES MAKEPART, CAUSES ERRORS	Closed	1	.	.	Acad: Libraries	CODE FIX	9/23/2009	9/23/2009	9/23/2009
1259156	TEST IGNORE: CRASH SOMETIMES STARTING INTENT	Closed	1	.	.	Acad: UI	CODE FIX	9/28/2009	9/28/2009	9/28/2009
1259809	CAN NOT GET INTENT 2010 LICENSE (STANDALONE)	Closed	1	.	.	Core: Installation & Licensing	CODE FIX	9/29/2009	9/29/2009	9/29/2009
1273769	INTENTUP TRANSLATES ~N IN STRING TO UNPARSABLE NEWLINE	Will Not Fix	1	.	.	Core: IntentUp	AS DESIGNED	11/4/2009	11/4/2009	11/4/2009
1273773	REF DOESNT HANDLE MULTIPLE COLONS IN REFCAINS	Closed	1	.	.	Core: Kernel	CODE FIX	11/4/2009	11/4/2009	11/4/2009
1071584	MULLDESIGN OF DRAWING DOESN'T CLOSE/REMOVE THE DWG FILE	Closed	2	SP1	.	Invr: Drawings	CODE FIX	4/23/2008	4/23/2008	4/23/2008
1118360	REGRESS-CAN'T CREATE CENTERLINE FROM CENTERMARKS.	Closed	2	.	.	Invr: Drawings	CODE FIX	9/7/2008	9/7/2008	9/7/2008
1148919	REPEAT NEW WILL CAUSE AUTOMATION BROWSER IN BAD STATUS.	Will Not Fix	2	.	.	Invr: General	NOT REPRODUCIBLE	10/29/2008	10/29/2008	10/29/2008
1204552	IVPARTEXTRUDE FAILS IN 2010, BUT WORKS IN 2009	Closed	2	.	.	Invr: Features	CODE FIX	2/27/2009	2/27/2009	2/27/2009
1218731	IASSEMBLYEXTRUDEFEATURE DID NOT WORK AS DESIRED (RUN OUT OF MEMORY)	Will Not Fix	2	.	.	Invr: Assembly	Back to Tech	5/4/2009	5/4/2009	5/4/2009
1239464	ERROR ON SAVE AFTER CHANGING DYNAMIC RULES	Will Not Fix	2	.	.	Invr: General	Back to Tech	7/15/2009	7/15/2009	7/15/2009
1239501	EXTRUSION FAILS FIRST TIME BUT THEN WORKS	Closed	2	.	.	Acad: General	CODE FIX	7/15/2009	7/12/2009	8/16/2009

3.4 The following high-severity defects were addressed in Update 4:

DID	Summary	Status	Se...	Generic1	Generic2	Feature	SubStatus	Created
1227665	NLA AND RELATED NOT INSTALLED BY INSTALLER	Closed	1	.	.	Core: Installation & Licensing	CODE FIX	6/8/2009
1233659	NOT INSTALLING ON AUTOCAD MECHANICAL 2010	Closed	1	.	.	Core: Installation & Licensing	CODE FIX	6/15/2009
1236331	MULTILINE NOT REALLY SUPPORTED	Closed	1	.	.	Core: Immediate Window	CODE FIX	6/24/2009
1236939	CONSTRAINTS DEFINED IN INTENT MODEL BUT MISSING IN INVENTOR ASSEMBLY AFTER...	Will Not Fix	1	.	.	Invr: Assembly	AS DESIGNED	6/26/2009
1246106	FOR STATEMENT DOES NOT AUTO-DECLARE ITS COUNTER	Closed	1	.	.	Core: Kernel	CODE FIX	8/11/2009
1248602	ERROR 2 RETURNED DURING STARTUP ON SWEDISH XP	Will Not Fix	1	.	.	Core: Installation & Licensing	.	8/20/2009
1248672	ALLOW OPTION TO TURN OFF TIMESTAMPS IN SYSTEM MESSAGES	Closed	1	.	.	Core: API	CODE FIX	8/20/2009
1249298	MINHOST SHOULD REMEMBER THE SEARCH PATH	Closed	1	.	.	MinHost: General	CODE FIX	8/24/2009
1257235	EXTRUDESOLID USES MAKEPART, CAUSES ERRORS	Closed	1	.	.	Acad: Libraries	CODE FIX	9/23/2009
1259156	TEST IGNORE: CRASH SOMETIMES STARTING INTENT	Closed	1	.	.	Acad: UI	CODE FIX	9/28/2009
1259809	CAN NOT GET INTENT 2010 LICENSE (STANDALONE)	Closed	1	.	.	Core: Installation & Licensing	CODE FIX	9/29/2009
1273769	INTENTUP TRANSLATES ~N IN STRING TO UNPARSABLE NEWLINE	Will Not Fix	1	.	.	Core: IntentUp	AS DESIGNED	11/4/2009
1273773	REF DOESNT HANDLE MULTIPLE COLONS IN REFCAINS	Closed	1	.	.	Core: Kernel	CODE FIX	11/4/2009
1071584	MULLDESIGN OF DRAWING DOESN'T CLOSE/REMOVE THE DWG FILE	Closed	2	SP1	.	Invr: Drawings	CODE FIX	4/23/2008
1118360	REGRESS-CAN'T CREATE CENTERLINE FROM CENTERMARKS.	Closed	2	.	.	Invr: Drawings	CODE FIX	9/7/2008
1148919	REPEAT NEW WILL CAUSE AUTOMATION BROWSER IN BAD STATUS.	Will Not Fix	2	.	.	Invr: General	NOT REPRODUCIBLE	10/29/2008
1204552	IVPARTEXTRUDE FAILS IN 2010, BUT WORKS IN 2009	Closed	2	.	.	Invr: Features	CODE FIX	2/27/2009
1218731	IASSEMBLYEXTRUDEFEATURE DID NOT WORK AS DESIRED (RUN OUT OF MEMORY)	Will Not Fix	2	.	.	Invr: Assembly	Back to Tech	5/4/2009
1239464	ERROR ON SAVE AFTER CHANGING DYNAMIC RULES	Will Not Fix	2	.	.	Invr: General	Back to Tech	7/15/2009
1239501	EXTRUSION FAILS FIRST TIME BUT THEN WORKS	Closed	2	.	.	Acad: General	CODE FIX	7/15/2009

3.5 The following defects were fixed in Update 5:

- Data stack corruption with Return Statement.
- Full-width Roman digits not parseable (Inventor - Japan)

3.6 Update 6

Update 6 fixes several problems that were identified in prior releases. This section describes those changes. Since Update 4 was released, the Intent team has transitioned to a new defect-tracking system. As a result, defect numbers have been reset, and begin with an INT- prefix to distinguish them.

3.6.1 INT-10: Inventor Adoption stops working

This problem was traced to a thread timing issue with the initialization dialog. The dialog has been removed and the output is directed to the status bar.

3.6.2 INT-11: Unhandled exception in Immediate pane

This problem occurred when the cursor was placed inside the prompt string.

3.6.3 INT-14: Unhandled exception in Design Editor / single-line IF THEN ELSE

This problem was traced to a compiler defect. Single-line IF THEN ELSE statements are permitted by the language, but a case was omitted in the compiler which handled them. This is also reported as defect ID 1260138.

3.6.4 INT-17: Render? not properly handled in AutoCAD

It was discovered in AutoCAD that setting Render? to false on a node in the assembly tree did not halt rendering for all children under that node. This problem was traced to a change made for Inventor, which handles rendering of Intent objects differently than AutoCAD. This was fixed by customizing the AutoCAD handling of Render? to match its original behavior.

3.6.5 INT-20: Design Editor exception with large Design sets

Extremely large Design sets, on the order of thousands, caused the Design Editor to crash Inventor. The problem was traced to resource exhaustion for icons, which were not being properly re-used.

3.6.6 INT-22: Commented-out underscore (Continuation) causes exception

If a statement with a continuation marker (an underscore character as the last one on the line) was partially commented out, an unhandled exception would occur. This was traced to the compiler properly handling the syntax, but the source-management tools believing the line was a continuation line, and thus getting out of synch.

3.6.7 INT-24: Sheet Metal adoption creates syntax error

When Sheet Metal parts were adopted which contained the double-quote character as a units designation in their style settings, the adoption produced improper syntax for a string.

3.6.8 INT-26: Unhandled exception in Design Editor for “Unresolved Reference”

Certain classes of errors were not being properly handled in the Design Editor. Correct code would function properly and return back to the Design Editor after updating, but certain errors during update were not handled and crashed the Design Editor when the error was passed back.

3.6.9 INT-30: Unhandled exception in Design Editor for “Undefined CFunc”

In another case of the Design Editor not properly handling “normal” Intent errors, this one occurred when a CFunc was properly declared, but not found in the dll it was declared to be in. The code has been changed to catch ALL Intent exceptions, not just certain ones, so the Design Editor should be much more robust in the face of normal Intent errors.

3.6.10 INT-32: Inventor AEC Exchange button greyed out

Also affected the “Convert to Weldment” button. An explicit reset was added to ensure these buttons were enabled.

3.6.11 INT-34: Return statement not working inside Child Rule parameter formulas

This problem was also reported as defect ID 1262189, and reported resolved in INT-7, but the specific case of Return inside of a Begin/End block for a Child Rule parameter formula was not detected. This case is created by IntentUp when `@{ }` syntax was used in Child rules.

4 Known Issues

A few issues have come to light which could not be rectified before release. These issues are listed in this section.

4.1 AutoCAD Viewports

There are known issues with the Viewport design. When a Viewport created by Automation is saved in a drawing, it is not properly marked as “owned” by Automation, and this causes it to be not updated when the drawing is subsequently re-opened. This causes a series of issues, including:

- Duplicated viewports. Automation creates a new Viewport directly over the saved one. This is not easily detectable.
- “Confusion” between viewports. AutoCAD sometimes seems to not correctly place geometry into duplicated viewports.

Possible workarounds:

- Don't use Viewports.
- Don't save drawings with Viewports created by AutoCAD. Instead, plot directly, write DWF files, or otherwise use the drawing without saving.
- Delete all viewports or layouts when a drawing is opened. This is only viable if you are only using Automation-created viewports.

This issue is being actively addressed, and should be corrected in a future release.

4.2 IntentUp is Limited to 32-Bit

IntentUp will only run on a 32-Bit system. This restriction will not be lifted, because the conversion process relies on 32-bit only legacy compiler code.

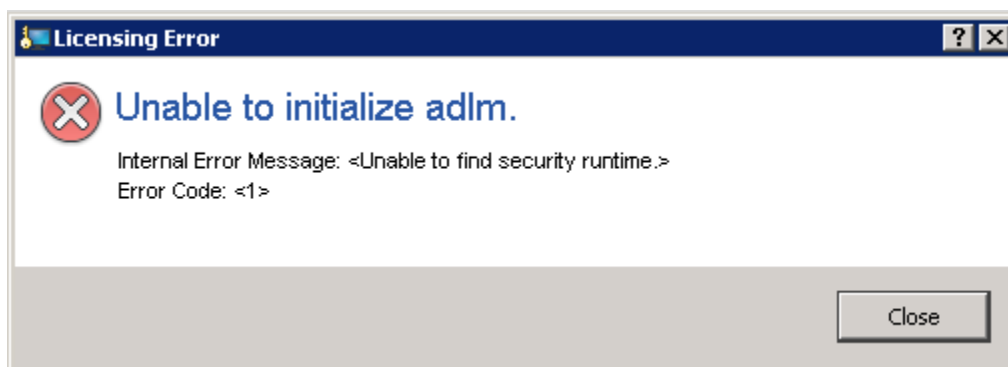
4.3 AutoCAD Ribbon support is turned off by default

Intent's additions to the AutoCAD ribbon panel are off by default. This means that there is no ribbon panel access to a new install of Intent on AutoCAD. The following command-line commands can be used to access Intent functionality on AutoCAD:

HINT_ABOUTBOX	Puts up Intent About box
HINT_DESIGNEDITOR	Puts up Design Editor
HINT_MODELBROWSERON	Puts up Model Browser Pane
HINT_MODELBROWSETOGGLE	Toggles Model Browser Pane
HINT_MODELBROWSEOFF	Takes down Model Browser Pane
HINT_DUMPXREC	Appears to dump IKM file contents, not sure about binary
HINT_HELP	Puts up Intent Help document
HINT_INITIALIZE	Initializes Intent
HINT_RIBBON	Turns on Ribbon changes
HINT_STARTUP	Called at startup (by acad.lsp) to initialize UI. Does not initialize Intent. Loads the Autodesk.Intent.AutoCAD.Addin.dll into the current process.
HINT_UPDATESTOGGLE	Toggles automatic updates
HINT_UPDATESON	Enables automatic updates
HINT_UPDATESOFF	Disables automatic updates
HINT_OPENIKP	Puts up File Dialog set to *.ikp

4.4 Intent 2010 on Inventor 2009 on Windows Vista x64

A problem was discovered after release of 3.0 for a particular combination of environments: Intent 2010, Inventor 2009, and Windows Vista x64 (and possibly on Vista x86, but this has not been verified). When Intent is initialized, the following dialog appears:



This was traced to a path configuration problem. This problem was not resolved in time for Update 1, so the following workaround is available.

Add Intent's "bin" directory to the system environment PATH variable.

4.5 Deinstallation Prevents Older Version from Starting

The same GUIDs are used for the Intent-Inventor starter addin. When Intent 2010 is uninstalled, the COM registration for any older addin must be performed manually. For example, if you want switch back to Autodesk Inventor Professional 2009, use the following instructions:

For 32-bit system:

- Cd C:\Program Files\Autodesk\Inventor Automation Professional 2009\Bin
- C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\regasm.exe Intent.Inventor.Addin.dll /codebase /tlb
- C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\regasm.exe Autodesk.Intent.Inventor.StarterAddin.dll /codebase /tlb

For 64-bit system:

- Cd C:\Program Files\Autodesk\Inventor Automation Professional 2009\Bin
- C:\WINDOWS\Microsoft.NET\Framework64\v2.0.50727\regasm.exe Intent.Inventor.Addin.dll /codebase /tlb
- C:\WINDOWS\Microsoft.NET\Framework64\v2.0.50727\regasm.exe Autodesk.Intent.Inventor.StarterAddin.dll /codebase /tlb
-

4.6 Installation Issues

4.6.1 Installation Links do not work

The Documentation and Support links at the bottom of the installer windows do not connect to the proper content. In some cases, they refer to AutoCAD content. This problem is due to missing files. It will be corrected in a future release.

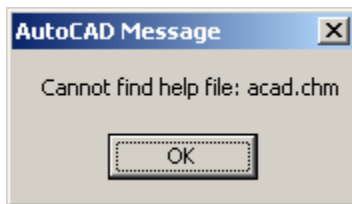
4.6.2 Install Tools and Utilities does not work

At the Launch window, the “Install Tools and Utilities” option takes the user to a page which has grayed-out Network License Manager and Network License Activation Utility check-boxes. The stated reason is “This product cannot be installed on the current operating system.” This problem is due to a misconfiguration. Depending on the utility, it may have been installed or may not be present.

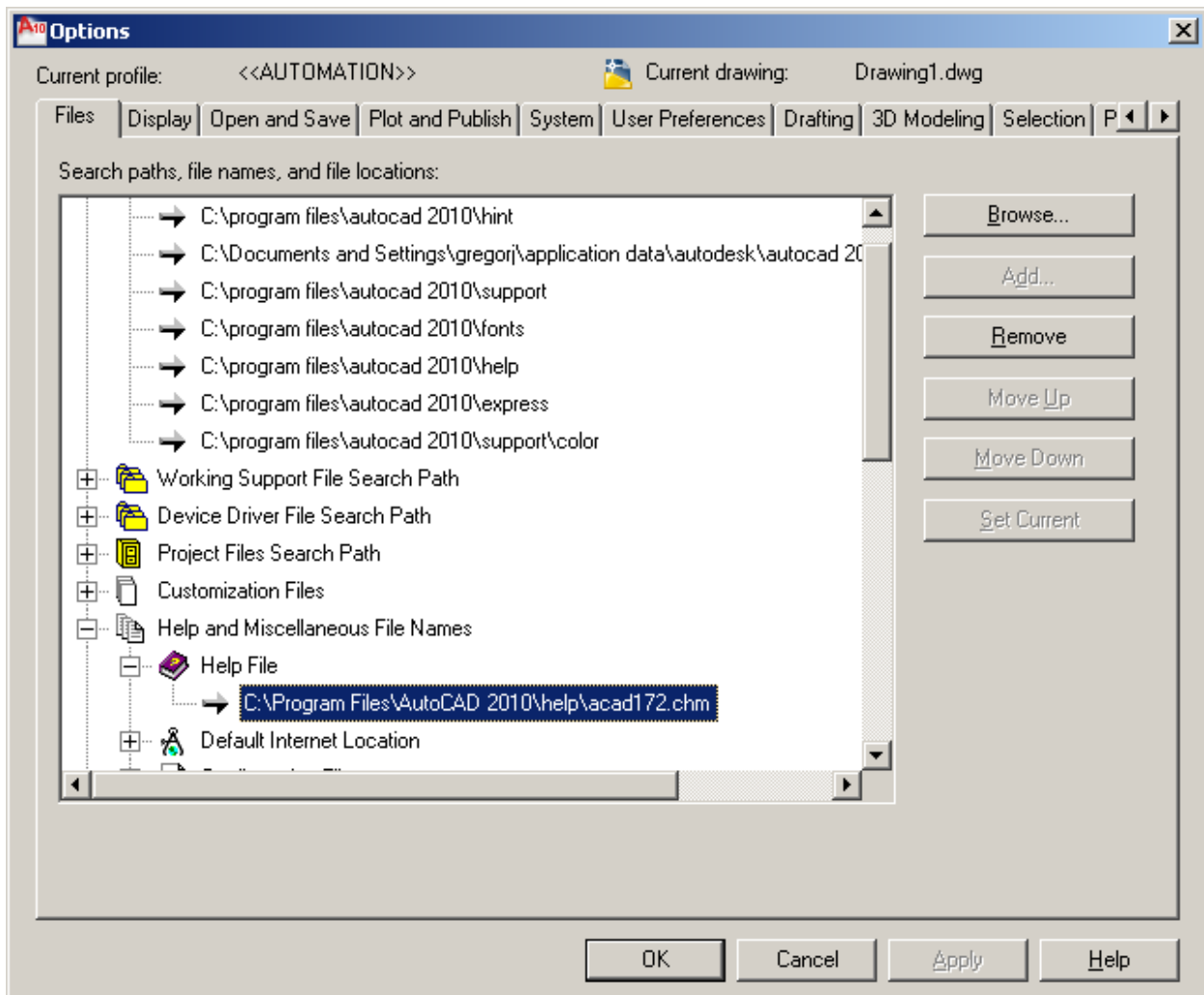
If these tools are required, contact Intent Support.

4.7 AutoCAD Help does not open

The Intent Profile for AutoCAD that gets installed is out of date. Certain portions of it refer to locations that are for AutoCAD 2009. One important symptom of this is that AutoCAD’s help is not found when the F1 function key is pressed, with the error as shown:



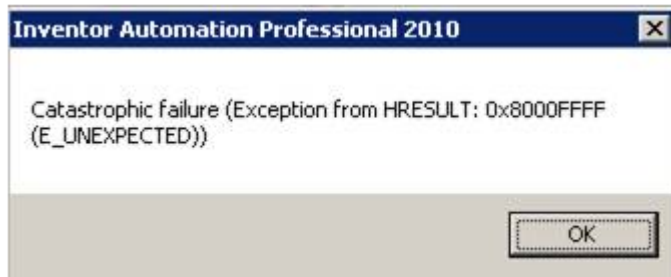
To correct this problem, open the profile using the Options command, and navigate to the Files tab, as shown:



Change acad172.chm to acad180.chm (click Browse..., select the acad180.chm file, and click Open), and F1 Help should be restored. There may be other manifestations of this problem; comparing the default profile to the <<Automation>> profile will be enough to determine what changes need to be made.

4.8 Sheetmetal Flat Pattern Adoption

Due to a defect in Inventor 2010, attempting to change the “has flat pattern” state of an adopted Sheetmetal part will generate a “Catastrophic Error” dialog if the hasFlatPattern? parameter is changed from True to False.



This is actually not as “catastrophic” as it appears. After selecting OK, work can continue, and nothing is permanently affected. The part file does not contain the flat pattern, and when subsequently used, the error does not re-appear.

As a workaround, two factory files can be used, one containing a flatpattern, and one without, and invoked according to whether the flat pattern is needed or not.

4.9 Data Corruption with Update 4

Update 4 (build 462) contained a flaw that causes data stack corruption when the Return statement is used in some cases. Update 5 (build 465) corrects this problem.

4.10 Instability in Japanese AutoCAD

As of Update 6: We are investigating certain problems with Intent running inside the Japanese version of AutoCAD. At this time, no fix is available for this problem.