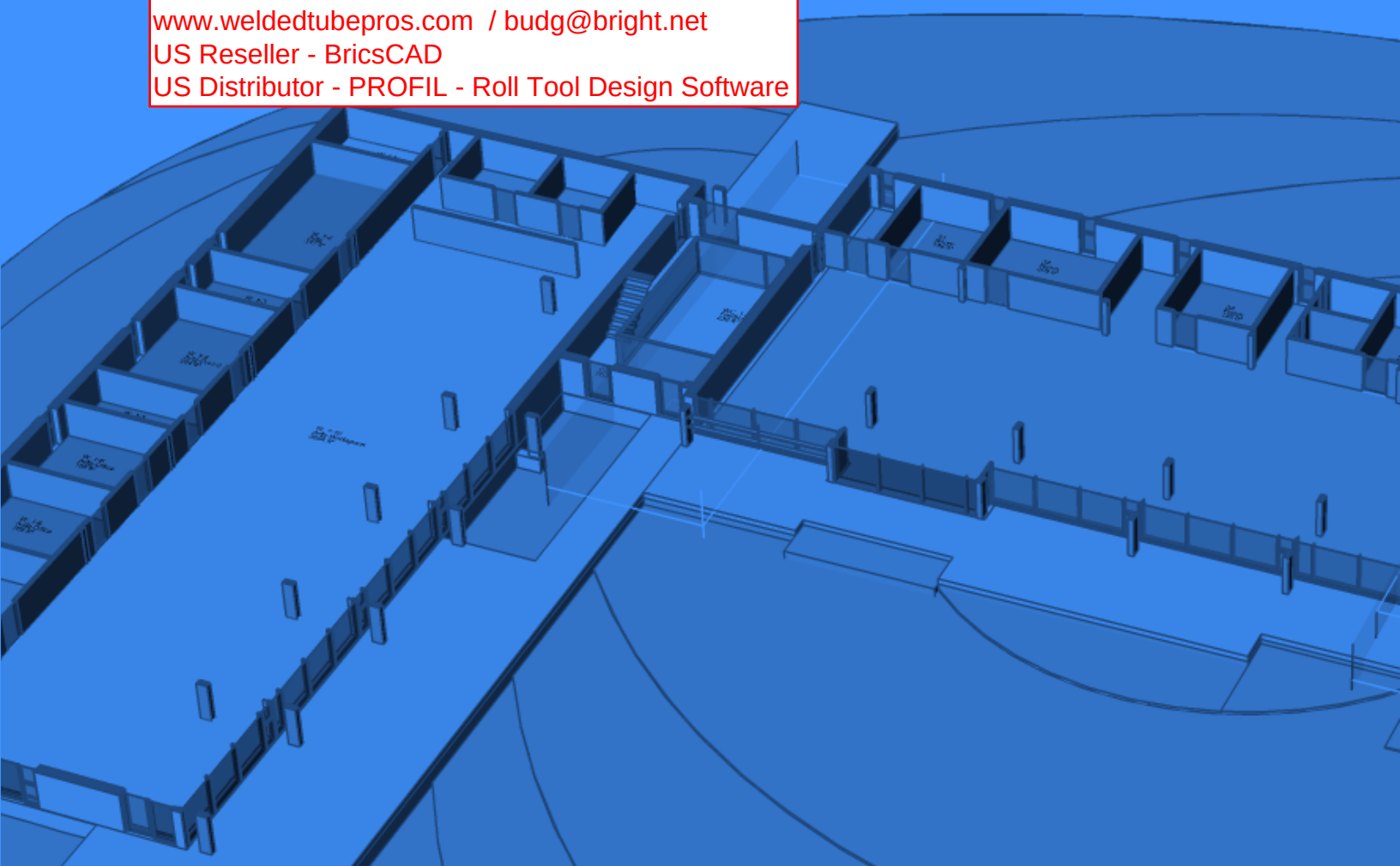
UPDATED FOR V17

BRICSCAD®

FOR AUTOCAD® USERS

Comparing User Interfaces
Compatibility of Drawing Elements
Customizing and Programming BricsCAD
Operating Dual-CAD Design Offices
Working in 3D
BIM, Sheet Metal, & Communicator



Payment Information

Suggested Price
US$33.40

By Email
Acrobat PDF format:
Allow for a 15MB download.

PayPal
To pay by PayPal, send payment to the account of grabowski@telus.net at https://www.paypal.com/.

Use this easy link to pay:
https://www.paypal.me/upfrontezine/33.40

PayPal accepts funds in US, Euro, Yen, Canadian, and 100+ other currencies.

Check or Money Order
We can accept checks in the following currencies:

- US funds drawn on a bank with address in the USA.
- Canadian funds drawn on a bank with a Canadian address (includes GST).

Make cheque payable to 'upFront.eZine Publishing'.

Please mail your payment to:
"BricsCAD for AutoCAD Users"
upFront.eZine Publishing, Ltd.
34486 Donlyn Avenue
Abbotsford BC
V2S 4W7 Canada

Visit the *BricsCAD for AutoCAD Users* Web site at http://www.worldcadaccess.com/ebooksonline/. At this Web page, editions of this book are available for BricsCAD V8 through V17. Purchasing an ebook published by upFront.eZine Publishing, Ltd. entitles you to receive the *upFront.eZine* newsletter weekly. To subscribe to this "The Business of CAD" newsletter separately, send an email to grabowski@telus.net.

Copyright Information

10th edition: Based on BricsCAD V17
First printing: 18 November 2016
Second printing: 11 January 2017

Technical writer: **Ralph Grabowski**

Table of Contents

APPENDICES

AUTOCAD-BRICSCAD DICTIONARY

BricsCAD's terms closely follow AutoCAD's jargon, but there are a few differences.

AutoCAD Term	BricsCAD Equivalent
ADS	SDS (software development system)
ARX	BRX (BricsCAD runtime extension) TX (Teigha runtime extension)
AutoLISP	LISP
Design Center	Drawing Explorer
Implied intersection	3dIntersection
Intersection	2dIntersection
Macros	Tools
Model documentation	Generative drafting
Navigation Cube or ViewCube	Lookfrom widget
Object	Entity
Options	Settings
Osnap	Esnap (entity snap)
Palette	Panel (bar)
Shortcut menu	Context menu
Xdata	EED (extended entity data)

CHAPTER ONE

BricsCAD for AutoCAD Users

THIS BOOK HELPS YOU MAKE THE TRANSITION FROM AUTOCAD® TO BRICSCAD. HERE YOU learn about the benefits of using BricsCAD while saving your firm a lot of money on software expenditures. You'll read about the advantage to switching to BricsCAD, how similar it is to AutoCAD, and the transition issues on which to keep an eye.

We provide you with detailed information on issues like the differences and similarities in user interfaces between the two CAD programs, compatibility of DWG files, and even how to operate two CAD systems in your design office.

At the end of the book, we provide you with useful appendices that exhaustively cross-reference command and system variable names between the two CAD systems — along with alias names, shortcut keystrokes, and mouse button actions.

BricsCAD V17 for AutoCAD Users is meant for you if you are

- An AutoCAD user considering switching to BricsCAD
- A CAD manager adding licenses of BricsCAD to complement your AutoCAD shop
- A design firm working with clients using one CAD package or the other

Or perhaps you are simply wondering about the differences between market leader AutoCAD and aggressive up-and-comer BricsCAD. Whichever the case, this book is for you. Now in its 10th edition, the book is updated to include functions added to BricsCAD V17.

Welcome!

The Bricsys Benefit

Bricsys is a small company compared to Autodesk, whose executives have grown the company into a two-billion-dollar-a-year enterprise. But dealing with a firm as enormous as Autodesk carries a with certain amount of risk. It pays to be aware of the risks.

THE AGONY OF AUTOCAD

Autodesk offers a rich variety of over a hundred software packages and bundles. AutoCAD itself comes in twenty variations, such as versions specific to architecture and civil engineering, or with two or more programs bundled together, such as Product Design Collection. This much choice can become confusing for potential customers determining which product or bundle is best for their need.

Given a large number of CAD programs that depend on the good will a single software company has a risk. When software crucial to the operation of your company might become a drag on profits, a large company might stop supporting it. Autodesk fine-tunes its products to maximize profits on behalf of its shareholders, and so the software you buy today may not be available tomorrow.

For example, Autodesk in years past has moved customers of its FM:desktop facilities management software to another company; halted development of its Constructware construction management software; orphaned users of other packages, such as Generic CADD (a low-cost CAD package), Actrix Technical (diagramming software), StudioDesk (architectural concept software), Mechanical Desktop (AutoCAD-based 3D mechanical design software), 123D.com, and Impressions (post-design rendering software) — among others.

Being a large company, Autodesk needs to charge prices that tend to be high. The $4,200 price of its foundation drafting package, AutoCAD, is 4x to 10x more costly than many office productivity packages. Pricing AutoCAD high is just the start: the company's previous CEO famously boasted to financial analysts that her company could make up to 10x more money when customers moved from AutoCAD to 3D modeling software. The 10x increase comes out of your pocket.

Top products

AutoCAD	AutoCAD MEP (US Site)	Navisworks Products
AutoCAD Civil 3D	AutoCAD Plant 3D (US Site)	Revit
AutoCAD Electrical (US Site)	3ds Max	Simulation products (US Site)
AutoCAD Electrical iPad app	Alias Products (US Site)	Smoke (US Site)
AutoCAD LT	Inventor Products	Vault products (US Site)
AutoCAD Map 3D (US Site)	Maya	

All other products (US Site)

A	B-H	I-N
123D Catch mobile and desktop app	Beast	Infrastructure Design Suite
123D Circuits Windows and web app	BIM 360	Infrastructure Map Server
123D Design mobile and desktop app	BIM 360 Field	InfraWorks 360
123D Make mobile and desktop app	BIM 360 Field iPad app	InfraWorks 360 iPad app
123D Sculpt+ mobile app	BIM360 Glue	Instructables
Advance Concrete	BIM 360 Glue iPad app	Instructables mobile app
Advance Steel	Building Design Suite	Inventor Engineer-to-Order
Alias Automotive (now Alias AutoStudio)	Buzzsaw	Inventor HSM
AutoCAD 360	Buzzsaw Professional	Inventor LT
AutoCAD 360 mobile and web app	Buzzsaw mobile app	Inventor Publisher
AutoCAD Architecture	CAiCE Visual Transportation Products	Inventor Publisher mobile viewer
AutoCAD for Mac (Canadian Site)	Character Generator	Kynapse (now Gameware Navigation)
AutoCAD Design Suite	Civil Engineering Data Translator	Lighting Analysis for Revit
AutoCAD Freestyle	Collaboration for Revit	Lustre
AutoCAD Inventor LT Suite	Composite	MatchMover
AutoCAD LT for Mac (Canadian Site)	Configurator 360 iPad and web app	Maya LT
AutoCAD Mechanical	Constructware	mental ray
AutoCAD OEM	Creative Market	Meshmixer
AutoCAD P&ID	Design Review	MIMI
AutoCAD Raster Design	DWF Writer	Moldflow
AutoCAD Revit LT Suite	DWG TrueView	Moldflow Design
AutoCAD Structural Detailing (Canadian Site)	DWG TrueConvert (see DWG Viewers)	MotionBuilder
AutoCAD Utility Design	Ember	Motion FX
AutoCAD WS (now AutoCAD 360)	Entertainment Creation Suite	Mudbox
A360 mobile, web, and desktop app	Fabrication Products	Nastran
A360 mobile for iPhone and iPad	Fabrication CADmep	Nastran In-CAD
Autodesk PLM 360	Fabrication CAMduct	Navisworks Freedom
Autodesk PLM 360 mobile and web app	Fabrication ESTmep	
Autodesk University 2014 mobile app	Factory Design Suite	
AutoSketch	FeatureCAM	
	FBX	
	FBX Review mobile and desktop app	
	Flame	
	Flame Premium	
	Flare	
	Formit mobile and web app	
	Freewheel	
	Fusion 360	
	GIS Design Server	
	Helius Composite	
	Helius PFA	
	Homestyler Interior Design mobile and web app	
	HumanIK	

Autodesk's offerings of software at
http://www.autodesk.ca/en/products-standard
as of November, 2016

Subscriptions. Autodesk made annual subscriptions mandatory for AutoCAD after January 31, 2016, and so it no longer sells more perpetual licences. (Except in Japan, which is strongly resisting subscription payments.) Autodesk has stated that it makes more from customers on subscriptions than on perpetual licenses — which means that your firm is paying Autodesk more to run CAD than it needs to. This is because Autodesk charges 1/3 of the software's old perpetual price as its annual subscription fee: after three years, you are paying more with subscriptions than with a single perpetual license.

If your firm cannot afford the subscription fee upon renewal — monthly, annual, bi- or tri-annually — such as in the midst of the next recession, then the Autodesk's software will stop working after 30 days, and your staff is no longer able to use it.

License Terms. Upon installing the software, customers must agree to onerous terms dictated by Autodesk in its software license. Many customers don't bother reading EULAs (end user license agreements) because the text is lengthy, and SOME SECTIONS ARE MADE EVEN MORE DIFFICULT TO READ THROUGH THE USE OF UPPERCASE LETTERS.

If you do read it, you may be shocked to learn that you are allowing Autodesk to send agents into your private home and business to search for unauthorized copies. Worse, the EULA makes it illegal for customers to travel outside their country with Autodesk software residing on their computer. Before getting on that airplane, you are required to erase AutoCAD from your computer. While Autodesk means this to protect regional sales, it is shortsighted of Autodesk to block its customers from taking part in the reality of today's globalized business.

O - Z
ObjectARX
Photo on ReCap 360
Pixlr
Pixlr desktop app
Pixlr Editor
Pixlr Express mobile app
Pixlr Express web app
Pixlr-o-matic mobile and web app
Plant Design Suite
PowerMill
PowerShape
PowerInspect
Product Design Suite
Quantity Takeoff
RealDWG
Remote
Revit LT (Canadian Site)
Scaleform
Screencast
Seek
Showcase
Simulation Products
Simulation Mechanical
Simulation Moldflow Communicator
SketchBook Express mobile app
SketchBook Ink mobile app
SketchBook mobile app
SketchBook Pro 2016
SketchBook Pro 7
Socialcam mobile app
Stitcher Unlimited
Streamline
Structural Bridge Design
Tinkercad
Vehicle Tracking
VRED
VRED Design
VRED Professional

THE BUSINESS OF BRICSYS

In contrast to Autodesk, Bricsys makes choice easy by offering just one software package in three levels of capabilities, along with two vertical add-ons. Compare the list below with the Autodesk list on the facing page:

BricsCAD Classic	Budget-priced 2D CAD software with limited 3D modeling
BricsCAD Pro	All of Classic, plus 3D direct modeling, rendering, generative drafting, and all APIs
BricsCAD Platinum	All of Pro, plus 3D constraints, mechanical assemblies, and access to add-ons
BIM add-on	Building information modeling and IFC connection
Communicator add-on	Standard and proprietary MCAD file format translation
Sheet Metal add-on	Sheet metal design and CAM system output

Communicator requires a Pro or Platinum license; BIM and Sheet Metal require Platinum licenses.

Here is a comparison of some of the major capabilities of each edition. For a more detailed comparison, please refer to https://www.bricsys.com/en_INTL/bricscad/compare/.

Function	BricsCAD Classic	BricsCAD Pro	BricsCAD Platinum
2D Design and Editing	Included	Included	Included
Printing, Exporting, Importing	Included	Included	Included
Constraints	...	2D	2D and 3D
ACIS 3D Solids Modeling	Viewing	Modeling, editing, viewing	Modeling, editing, viewing
Direct 3D Editing	Viewing	Modeling, editing, viewing	Modeling, editing, viewing
History-based 3D Modeling	Viewing	Modeling, editing, viewing	Modeling, editing, viewing
Design Intent	...	...	Modeling and editing
3D Assemblies and BOMs	Viewing	Viewing	Modeling, editing, viewing
Generated Drawings	Viewing	Included	Included
Surface 3D Modeling	Viewing	Viewing	Modeling, editing, viewing
Deformable Modeling	Viewing	Viewing	Modeling, editing, viewing
Kinematic analysis	Viewing	Viewing	Modeling, editing, viewing
BOM	Viewing	Viewing	Modeling, editing, viewing
GIS	Included	Included	Included
Rendering	...	Included	Included
Customization	Included	Included	Included
Programming	LISP, TX	LISP, TX, BRX, VBA, .Net	LISP, TX, BRX, VBA, .Net

BricsCAD Platinum is the full-featured version of BricsCAD: it has everything. The Pro version is almost identical, leaving out only the parametric-based 3D modeling and 3D constraints. The Classic version costs the least because it leaves out features for which Bricsys has to pay royalties to other software companies. This means that the Classic version excludes ACIS modeling and editing, and VBA, BRX, and .Net programming.

To read and write DWG and DXF files, BricsCAD uses the highly compatible Teigha libraries from Open Design Alliance.

In summary, Bricsys has a simple-to-understand product line, doesn't charge high prices, and doesn't impose mandatory subscriptions. The terms in its license allow you to use the software in any country, and Bricsys does not threaten to send agents into your home.

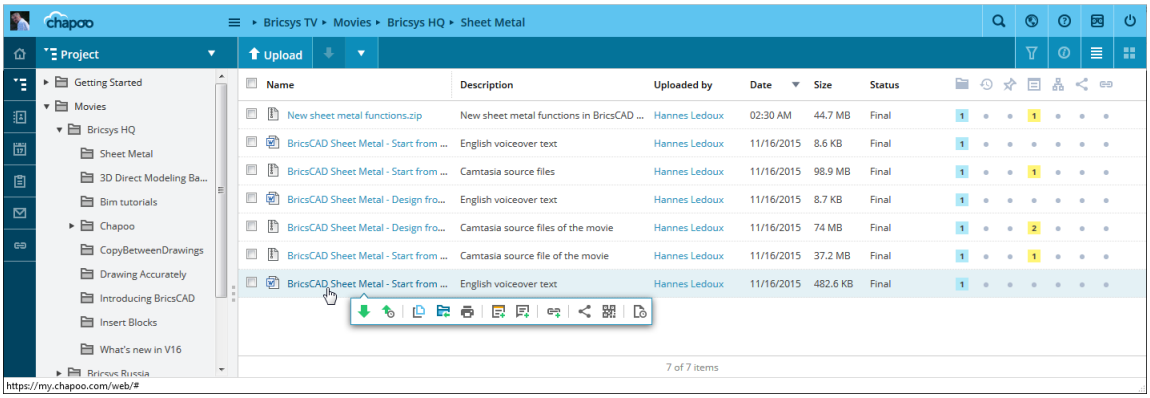
Chapoo Project Management

For managing drawing projects, Bricsys recommends Chapoo. This browser-based communication, collaboration, and project management system does not require BricsCAD, so it works with any office system. It is, nevertheless, integrated into BricsCAD through options in the File menu.

Chapoo project management offers your firm the following benefits:

- **Enjoy Speed.** View multi-megabyte drawings in seconds, zooming in on details and examining annotations with fly-over text that lists time stamp and author. Chapoo supports 70+ file formats, like Excel, Visio, MS Project, and AutoCAD.

- **Upload Files.** Drag and drop files into the upload area of Chapoo, and you're done.
- **Share Files and Folders.** Files can be shared through email, Facebook, or Twitter; folders are shared with other Chapoo users only. You have 1GB of online storage space to start with.
- **Create Annotations.** Drag a rectangle over the text or image to highlight, and then enter mark-ups in a few words or attach multiple text pages to the annotation. When you notify friends about it, Chapoo emails a link with direct access to the file with the annotations.
- **Manage Compliance.** Follow a continuous audit trail of the entire project process. Chapoo automatically maintains log files of project activities and participants.
- **Enjoy a Single Access Point**. You have a repository of all actions, documents, meetings, and participants in a single location.
- **No Software to Install.** Work with an ASP (application service provider) system. The software runs on central servers with guaranteed access 24/7; you only need an Internet connection and a supported Web browser.

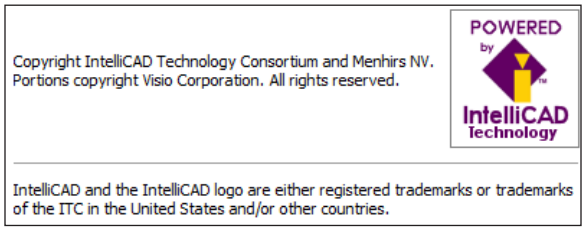


The Web interface while using Chapoo

There are two versions, the for-free Chapoo Free and the fee-based Chapoo. Unlimited access is available to an unlimited number of participants through a yearly flat fee based on industry type and company size. Portable versions of the service are available for Android and Apple tablets. For more information, please visit http://www.chapoo.com/en_INTL/.

BRICSCAD IS NOT INTELLICAD

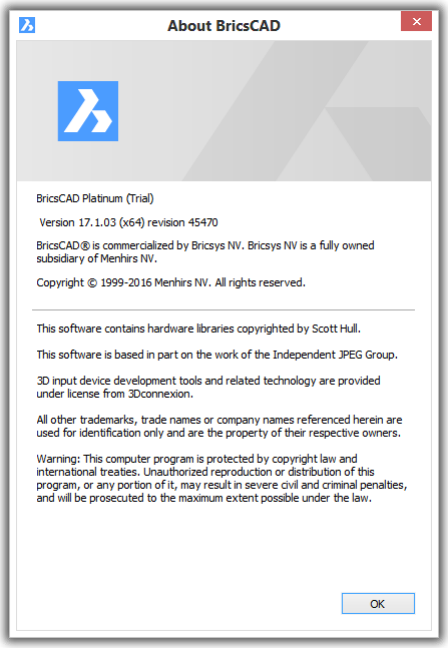
Readers familiar with BricsCAD may know it was based on IntelliCAD. *Was* is the important word here.



Old splash screen showing IntelliCAD logo

The very earliest versions of BricsCAD were rebranded releases of IntelliCAD, an AutoCAD workalike programmed by the IntelliCAD Technical Consortium. Bricsys at the time sold the software primarily in northern Europe.

(*A little history:* ITC was created in 1998 by Visio, now part of Microsoft, after deciding to leave the CAD market. Years earlier, Visio had purchased software named "Project Phoenix" from SoftDesk, who in the mid-1990s was the largest AutoCAD third-party developer. SoftDesk had began coding Phoenix after executives worried that Autodesk might cut off access to APIs. When Autodesk purchased SoftDesk, the US Federal Trade Commission required that it spin off Phoenix. Visio made the purchase and renamed it IntelliCAD, but then later spun it off to the ITC. The consortium continues to update IntelliCAD to this day, whose members rebrand the software for resale in their regions.)



Splash screen of today's BricsCAD

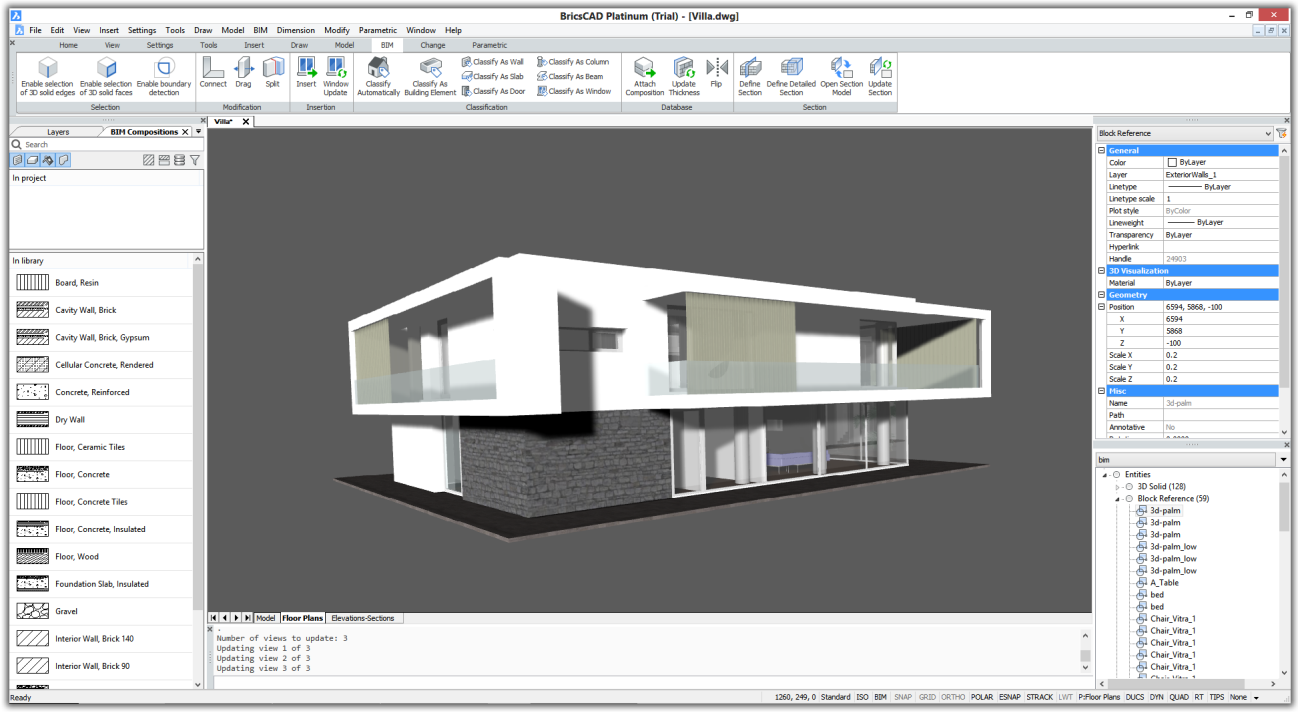
The executives of Bricsys decided they would rather develop BricsCAD on their own rate, faster than the ITC's pace, and made the decision to write all-new code. This huge undertaking took a couple of releases, beginning with V8. Bricsys contributed the new code to the ITC, which helped speed up improvements to IntelliCAD.

During BricsCAD V8 and V9, Bricsys concentrated on replacing all the ITC code with its own new programming code. As of BricsCAD V10, the software is 100% Bricsys, and so the About dialog box no longer mentions IntelliCAD or Visio.

With Bricsys' purchase in 2010 of the programming division of Russian software company LEDAS, functions grew dramatically with V12 and the following releases. Today we see BricsCAD equipped with 3D constraints, sheet metal and BIM modeling, 3D deformable and surface modeling, and many other functions not found in IntelliCAD.

The BricsCAD Advantage

BricsCAD shouldn't be considered just because it lacks the negative aspects of AutoCAD; it has advantages of its own. The benefits include a similar user interface, extra commands and variables, support for operating systems other than Windows, built-in direct 3D modeling and editing, 3D constraints, a no-charge developer network — and lower pricing.



BIM workspace user interface of BricsCAD running on Windows

NEAR-IDENTICAL USER INTERFACE

When you launch BricsCAD for the first time, you will notice that it looks very much like AutoCAD — complete with ribbons and/or toolbars, menu bar, command prompt, and palettes.

As illustrated amply by appendices at the back of this book, BricsCAD uses the same names for many AutoCAD commands, system variables, and aliases. It even has the same keystroke shortcuts. Commands that are missing are AutoCAD commands you probably weren't using anyhow, such as those for database linkages or 3D laser points.

The user interface of BricsCAD is available in English and a dozen other languages, and it can be customized. Chapter 2 describes the user interface in detail.

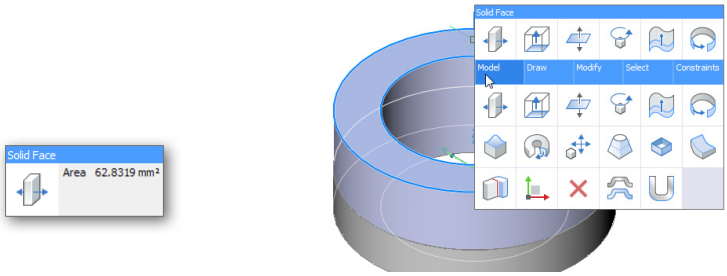
Extra Commands and System Variables

AutoCAD boasts more than 1,300 commands and system variables. BricsCAD mimics many of them; in addition, it has additional commands and variables that you will find useful yet are not found in AutoCAD.

For example, all entity (object) snaps in BricsCAD have their own command names, such as Intersection and Midpoint. BricsCAD has more ways to select objects than AutoCAD, with ones such as circular and external selection sets. There are commands for manipulating extended entity data (available in AutoCAD only through programming) and for editing 3D models and sheet metal designs directly (not available in AutoCAD at all).

Autodesk stores user settings in a number of locations scattered throughout AutoCAD, some of which can be difficult to access; some are unavailable, even as system variables. In contrast, BricsCAD summarized all variables and options in a single dialog box accessed by the Settings command. BricsCAD offers you extra control through variables known as "preferences," such as BkgColor for specifying the background color of the drawing area and CmdLineFontName for setting the name of the font used by the command bar.

Unique to BricsCAD is the Quad cursor. When you hover over a feature, such as a 2D object or a 3D face, it instantly reports information about. (You can specify which information is reported.) Move the cursor downwards, and the Quad display the commands most likely needed to manipulate the feature. When no objects are selected, a right-click displays the Quad cursor, which the presents you with drawing commands. You can customize the Quad cursor to your liking.



The Quad cursor provides fast access to entity data (left) and context-senstivie commands (right)

3D Direct Modeling and Constraints

When it comes to 3D design, BricsCAD is dramatically ahead of AutoCAD. The Platinum edition applies 3D constraints and infers design intent — in addition to placing 2D dimensional and geometric constraints. AutoCAD does not have 3D constraints or design intent. (The Classic and Pro editions of BricsCAD have 2D constraints.)

BricsCAD offers 3D modeling functions not found in AutoCAD, such as these 3D constratins and entity snaps

To model assemblies of complex products, BricsCAD employs *.dwg* files of mechanical components and orders them in hierarchical structures, even reading assembly structures from other MCAD systems, like Solidworks and Inventor. Kinematic analysis of moving and rotating parts reviews motions forwards and backwards in real time. Sheet metal and BIM (building information modeling) design are optional add-on modules. None of these are in AutoCAD or operate with .dwg files.

Direct modeling and editing lets you directly interact with 3D models. See chapter 6 for more. While this is possible in AutoCAD, Autodesk tells its users to use their stand-alone Fusion 360 software and pay an extra cost.

APIs and Customization

Bricsys is making it easier for third-party developers to adapt AutoCAD add-ons to BricsCAD — just as Bricsys works to make it easy for AutoCAD users to learn BricsCAD. For programmers, this is done through APIs, short for "application programming interfaces," and BricsCAD supports almost the same list of APIs as does AutoCAD.

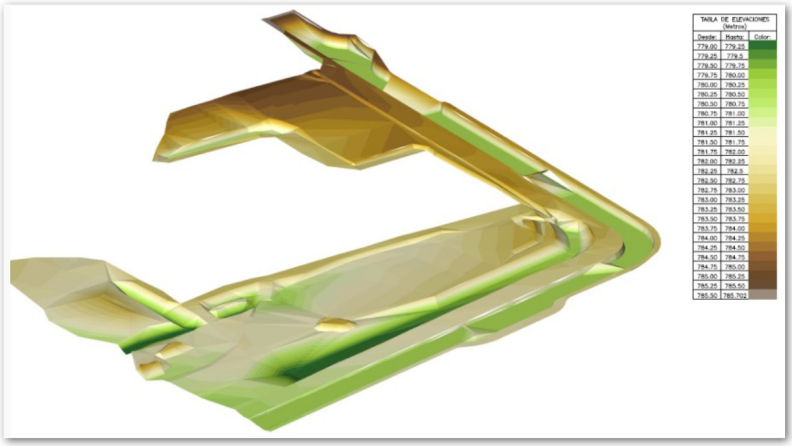
AutoCAD API	Equivalent in BricsCAD	Notes
Action Recorder (*)	Scripts, SCR	AutoCAD's Action Recorder scripts cannot be edited; scripts recorded by BricsCAD can be edited
ActiveX	ActiveX	In-place editing; not available in BricsCAD for Linux or Mac
ADS	SDS	ADS code ported from AutoCAD requires just a recompile using BRX headers; ADS/SDS are deprecated by Autodesk and Bricsys.
ARX	BRX or TX	Ported ARX code requires just a recompile using new BRX headers; when used with TX (ex-DRX), ported ARX code must be rewritten
AutoLISP	LISP	Ported AutoLISP code runs as-is in BricsCAD; no changes needed, includes support for Vl, Vlr, Vla, and Vlax functions and encryption
COM	COM	Ported AutoCAD COM code runs as-is in BricsCAD; not available in BricsCAD for Linux or Mac
CUI	CUI	Ported AutoCAD CUI files made need adjusting for BricsCAD
Diesel	Diesel	Ported Diesel code runs as-is in BricsCAD; no changes needed
DCL	DCL	Ported DCL code runs as-is in BricsCAD; no changes needed
.Net	Teigha.NET	BricsCAD provides Teigha.NET and extra BRX-managed wrappers; not available in BricsCAD for Linux, Mac, or Windows Classic versions
...	TX	Teigha eXtensions (formerly DRX) from Open Design Alliance; not available in AutoCAD.
VBA	VBA	Current AutoCAD VBA code runs as-is in BricsCAD for Windows; not available in BricsCAD Linux, Mac, or Windows Classic versions
VSTA	...	VSTA is unavailable in BricsCAD

Generally, BricsCAD provides a nearly identical subset of equivalent function names. In the case of non-compiled code, such as LISP and DCL, you just drop it into the BricsCAD environment. With compiled code, you recompile it using headers provided by Bricsys to registered developers. See https://www.bricsys.com/en_INTL/applications/developers/.

Examples of Add-ons

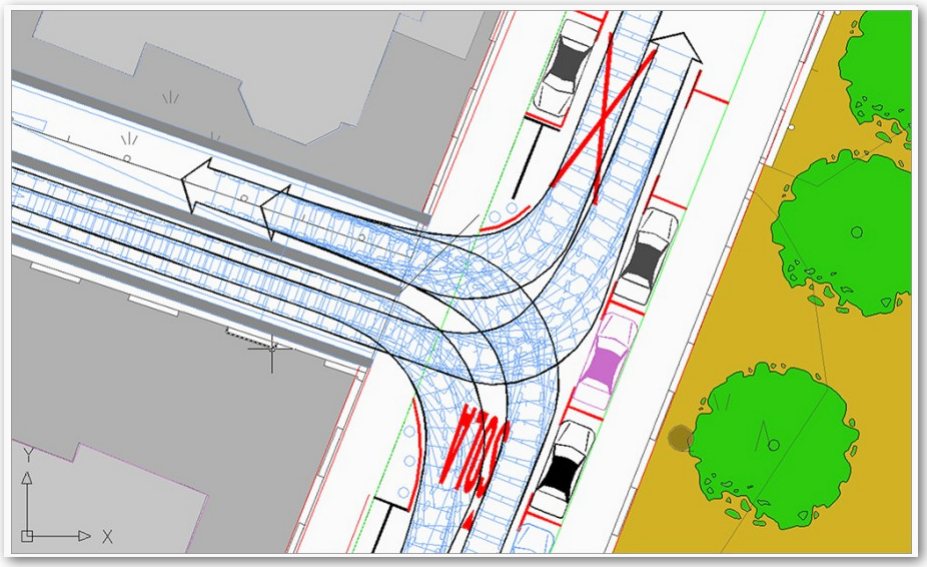
Independent programmers have written dozens of add-ons that tailor BricsCAD for specific applications in the areas of AEC, civil, data exchange, electrical, GIS, survey and mapping, general tools, HVAC, mechanical, packaging, rendering, and structural design. Here are a few examples:

DTCPRO from Disedig performs digital terrain modeling (TIN and contouring), cross-sections, longitudinal profiles, linear works, and volumetrics inside BricsCAD. http://www.disedig.com/Dtcpro.html



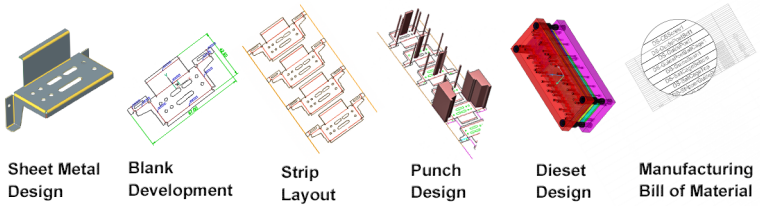
Color coding indicating height of terrain

Autopath from **CGS Plus** generates swept path analysis by analyzing maneuverability and clearance of steered vehicles of all types for intersections, roundabouts, and parking lots.
http://usa.cgsplus.com/Products/AutopathSweptPathAnalysis.aspx



Turning paths of large vehicles

DS Tools from Design Sense adds to BricsCAD's basic sheet metal capabilities with blank development, strip layouts, punch designs, dieset designs, manufacturing bills of material, and quotations.
http://www.thedesignsense.com/DSTools



Range of tasks performed by DS Tools

At time of writing, more than 400 applications are available for BricsCAD. For the complete list, visit the company's Applications Store at https://www.bricsys.com/applications/.

No-charge Developer Network

Bricsys does not charge third-party developers a fee; Autodesk charges an annual fee of $1,400 and up. You do not pay Bricsys a fee to join, you do not pay an annual membership, you do not pay for support, and you do not pay royalties on shipping products.

The reason support is free is because Bricsys feels that to become a successful CAD company it needs to encourage the development of many, *many* add-on applications — currently 1,500, a number that includes ones written privately. The company feels so strongly about third-party development that it has halted development of its own add-ons, except for a few that benefit many users.

Bricsys now concentrates on two tasks:

- Improving BricsCAD
- Adding to APIs

End users also benefit from APIs. (The application programming interface is the software link between CAD software and programming languages/compilers.) When a third-party developer requests an addition to the API, the added code becomes a new feature in BricsCAD that end users can employ.

SUPPORT FOR MULTIPLE OPERATING SYSTEMS

Bricsys was foresighted enough to write its BricsCAD code so that it is independent of operating systems. The company offers versions of BricsCAD that run natively on Windows, Linux, and MacOS.

AutoCAD runs on Windows and MacOS, but not Linux. The MacOS version has a significant problem, because it leaves out about a third of the functions found in the Windows version, yet the Mac version as expensive as the Windows version. (Autodesk lists the missing functions http://www.autodesk.com/products/autocad/compare/compare-platforms.)

BricsCAD, by contrast, boasts nearly all the same functions in all three OS versions, as shown by the comparison chart at http://www.bricsys.com/en_INTL/bricscad/editions/.

LOWER PURCHASE AND MAINTENANCE PRICING

Perhaps the most dramatic difference from AutoCAD is that the most expensive version of BricsCAD is **4x cheaper** than AutoCAD. To put the math another way, your office can be outfitted with four seats of BricsCAD Platinum instead of one seat of AutoCAD — and have money left over to buy another computer.

BricsCAD has a single upgrade price and a single maintenance price for all editions. List prices at time of writing are as follows:

List Price [1]	AutoCAD	AutoCAD LT	BricsCAD Platinum	BricsCAD Pro	BricsCAD Classic
Perpetual License [2]	"$4,200"	"$1,080"	$ 940	$ 630	$ 520
Maintenance [3]	...	...	$ 200/year	$ 200/year	$ 200/year
Subscription [4]	$1,400/year	$ 360/year	...	...	...

[1] US$ pricing for single-user license; price may be different in other currencies;. Lower pricing usually available for multi-seat purchases and networked versions; student-use licences are free. Prices as at 6 November 2016.
[2] Autodesk "perpetual" licence price no longer available, and is shown for illustrative purposes based on 3x subscription cost.
[3] Annual maintenance requires a one-time perpetual license purchase; includes advanced support and all upgrades.
[4] Only subscription licenses available from Autodesk as of 2106; price shown for annual renewals, other terms available.

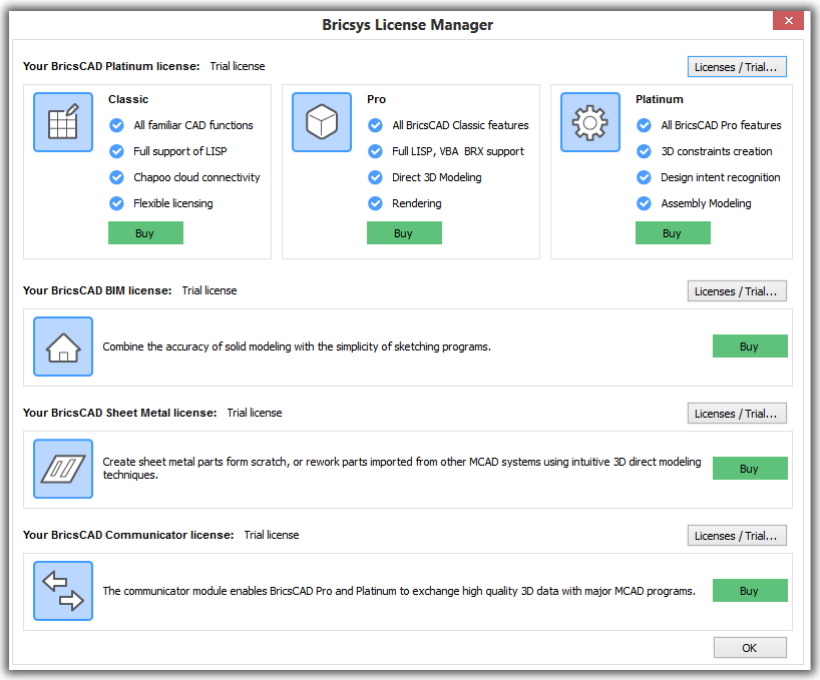
Autodesk eliminated all perpetual licenses during 2016 (except Japan). This means that annual subscription payments are compulsory when purchasing new licenses from Autodesk. After three years of subscription payments, you begin paying Autodesk more than with a single perpetual license payment. See http://www.autodesk.com/store for pricing details on all Autodesk products.

In contrast, BricsCAD saves you money through lower pricing to start off with, a maintenance fee that's lower than Autodesk's subscription cost, and allowing you to chose whether to upgrade (or not) or to subscribe (or not). You save even more money because BricsCAD has less stringent hardware requirements, and allows you to run on a free operating system, Linux (an option that is not available from Autodesk). See Chapter 5 for running CAD on Linux.

See https://www.bricsys.com/estore/ for pricing details on all Bricsys products.

Keep Your BricsCAD

If you like your old BricsCAD, you can keep your old BricsCAD. When new releases come out, Bricsys does not force you to give up your old software. When you get a license number for V17. it powers BricsCAD as far back as V14.



Licensing dialog box for BricsCAD V17

BricsCAD licenses can be moved between computers, just like AutoCAD. This lets you install the software as many times as you need, then just deactivate the current one to activate BricsCAD on another computer.

It Makes More than Cents

You could ask, "Are AutoCAD's additional functions worth the $3,175 difference in price?" For some users, a high price makes sense to them. But others will work out the benefit of the difference: "I can get things like 3D mesh modeling with Rhino at $1,000, add a Rhino-BricsCAD file converter ($95) — and still be two thousand dollars ahead."

For example, you could model a 3D boat hull in Rhino and then add 2D details and annotations with BricsCAD. Rhino is available from Robert McNeel & Associates at http://www.rhino3d.com/download; the 3DM converter is sold at the Bricsys eStore.

ALL ABOUT BRICSCAD BULK LICENSES

by Jason Bourhill

Once your firm has more than ten seats of BricsCAD, you should consider a bulk license for convenience and possible cost savings. The Bricsys bulk license system carries out unattended installs, configurations, and uninstalls of BricsCAD by the IT manager, as well as providing flexibility to end users. Bricsys offers two forms of bulk license, volume and network. Autodesk does not offer such licensing for AutoCAD LT.

VOLUME LICENSES

Volume licensing uses a single authentication key that is valid for a specific number of installs, as identified in the license agreement. After the software is installed, each user needs to activate their license online (requires an Internet connection) with the licensing server hosted by Bricsys.

Volume licenses suit firms whose staff that require continuous access to BricsCAD. The cost is the same as for individual All-In seat license.

NETWORK LICENSES

Network licensing uses a single authentication key that is valid for a specific number of simultaneous users, as identified by the license agreement. The license server is customer-hosted, and only the customer-hosted license server needs to activate licenses online with Bricsys. This means that none of the client computers need an Internet connection, which some firms prefer for security. Bulk license installation is usually undertaken by the firm's IT manager.

Network users have the option to book out a license, allowing for continued use away from the license server. This may incur additional cost.

Network licenses suit organizations whose staff require only intermittent access to BricsCAD. The cost is initially greater than with volume licenses; however, the ongoing costs can be significantly less. The key is the low threshold: network licenses can start from just one license and then grow from there, instead of the minimum of ten needed for volume licensing. Once the number reaches five, a discount becomes available on purchasing licenses. With larger numbers, a discount is also available for annual subscriptions.

SUPPORT FOR NETWORKS AND LICENSES

To download the network license manager, follow this link: https://www.bricsys.com/bricscad/tools/Bricsys-NetworkLicenseManager.msi.

Follow the advice of BricsCAD's online help when setting up the network by going to https://www.bricsys.com/en_INTL/support/ and then entering "network" as the search term. In addition, the Bricsys Knowledge Base covers typical network installation issues and error codes at https://www.bricsys.com/en_INTL/support/#85.

Information on how to use the Bricsys network license on a client computer: https://www.bricsys.com/bricscad/help/en_US/V17/BricsCAD/index.html?page=source%2FNetwork.htm. Large organizations may want to automate deployment through silent installation (Windows only): https://www.bricsys.com/bricscad/help/en_US/V17/BricsCAD/index.html?page=source%2FSilent_Installation.htm.

BricsCAD uses the Reprise license manager. For detailed information on the license manager software, download the PDF manual from the Reprise site: http://www.reprisesoftware.com/RLM_License_Administration.pdf. License administrator and user FAQs from Reprise Software: http://www.reprisesoftware.com/publisher/license-management-faq.php.

WHAT'S MISSING FROM BRICSCAD?

BricsCAD doesn't have every feature found in AutoCAD. As I update this ebook each year, the list becomes shorter with each release of BricsCAD. Here it is as of V17:

AutoPublish	CAD standards	Database links	Dynamic blocks*
Markups	Quick view thumbnails	PDF editing	Point clouds
3D mesh modeling			

*) BricsCAD edits these entities created in AutoCAD, although it cannot create dynamic blocks.

Chapter 3 provides complete details of which AutoCAD entities work in BricsCAD, which work partly, and those few that don't work at all.

At first glance, there are features in AutoCAD that appear to be missing from BricsCAD, but another glance shows that BricsCAD has near-equivalents operating under other names. Here are some examples:

AutoCAD Feature	BricsCAD Equivalent	Command Names in BricsCAD
Action Recorder	Script recorder	RecScript, StopScript
DesignCenter	Drawing Explorer	Explorer
QLeader	Leaders	DimLeader
Real-time dimensioning	Dimensioning with Quad	...
ViewCube	LookFrom widget	LookFrom
VSTA	VBA and .Net	VBA, AppLoad

What's Missing from AutoCAD

BricsCAD Platinum V17 offers these 2D and 3D functions that are not found in AutoCAD, which costs 4x as much:

- Placing 3D constraints
- Assembling parts into large models
- Inferring design intent
- Editing 3D models directly
- Analyzing kinematics
- Designing sheet metal
- Designing BIM
- Editing interactively with the Quad cursor
- Entering object snaps as command names
- Making circular, external, and other types of selection sets
- Manipulating extended entity data easily
- Accessing all system variables and options through a single dialog box
- Setting additional variables, such as BkgColor (specifies drawing area background color) and CmdLine-FontName (sets the font for command bar text)

System Requirements

Your IT department will appreciate that BricsCAD does not require expensive hardware or the latest operating system requirements to perform well. This is significant for these reasons:

- Design firms can often run BricsCAD on computers they already have. This extends the investment in hardware, and manages costs when they do upgrade
- BricsCAD uses less RAM and requires less CPU speed than AutoCAD, meaning more memory space and CPU power is available for users

RECOMMENDED HARDWARE

Autodesk and Bricsys recommend that your computer meet the following specifications. AutoCAD cannot run on smaller computers (like netbooks) whose screens have a resolution of below 1024x768. BricsCAD runs well on older computers. AutoCAD for Mac will not run on unsupported Apple computers; BricsCAD works well with older Macs.

Here are the recommended specifications for 64-bit systems.

Hardware	AutoCAD	BricsCAD
CPU	1.0GHz	1GHz or faster CPU
MacOS	Apple Mac Pro 4,1 or later MacBook Pro 5,1 or later iMac 8.1 or later Mac mini 3.1 or later MacBook Air 2.1 or later MacBook 5.1 or later	Any recent Mac
Minimum RAM MacOS	4GB 3GB	256MB, plus RAM required by OS
Recommended RAM MacOS	8GB or more 4GB	1GB or more
Hard Disk Space	6GB for installation	250MB for program files + 1GB free space
MacOS	3GB recommended	
Monitor Resolution	1024x768 minimum 1600 x 1050 recommended	1024x768 with true color (minimum)
MacOS	1280x800 minimum 2880x1800 recommended	1024x768 with true color minimum
Graphics Board	DIrectX 9 or 11 128MB (minimum) workstation-class Pixel Shader 3.0 or greater for 3D Direct3D for 3D For supported brands, see http://usa.autodesk.com/adsk/servlet/syscert?siteID=123112&id=18844534	Any XGA or better graphics board, such as from Intel, nVidia, and AMD Uses Redway3D for rendering
MacOS	Built-in graphics	Built-in graphics
Pointing Device	Mouse	Mouse
MacOS	Apple or Microsoft mouse or trackpad	Mouse or trackpad

Supported Operating Systems

Bricsys supports BricsCAD running on several dialects of Linux, as well as on MacOS (the new name for MacOS), and older releases of the Windows operating system.

Autodesk has not announced a Linux version, and no longer supports Windows Vista. While Autodesk has a version of AutoCAD for the Mac, it is missing numerous commands and most APIs.

Here is the list of operating system on which both CAD systems can run:

AutoCAD	BricsCAD
...	Windows Vista with service pack 2
Windows 7 SP1	Windows 7
Windows 8.1	Windows 8 or 8.1
Windows 10	Windows 10
MacOS v10.9 or later	MacOS v10.9 or higher
...	Ubuntu LTS Linux
...	Fedora Linux
...	OpenSuse Linux
...	Linux other distributions

The Windows versions of AutoCAD require Internet Explorer for functions such as help; BricsCAD works with any Web browser.

For more information on that operating systems on which BricsCAD runs, see http://bricsys.com/en_INTL/support/#30a=65

Information about AutoCAD running on the Windows operating system: https://knowledge.autodesk.com/support/autocad/troubleshooting/caas/sfdcarticles/sfdcarticles/Operating-system-compatibility-for-AutoCAD-and-AutoCAD-LT.html. AutoCAD for Mac operating systems: https://knowledge.autodesk.com/support/autocad-for-mac/troubleshooting/caas/sfdcarticles/sfdcarticles/Operating-system-compatibility-for-AutoCAD-for-Mac.html.

Just as you can try out AutoCAD free for 30 days, you can install and run the Platinum edition of BricsCAD for 30 days at no charge from http://www.bricsys.com . You can test the Linux, Mac, and Windows versions. Only Microsoft's VBA is disabled in the trial version. The size of the BricsCAD download file is 240MB, 9x smaller than AutoCAD's 2.2GB download file.

IN SUMMARY, BricsCAD operates much like AutoCAD — yet is much more economical.

———————

In the following chapters, we delve deeper into the themes sketched out by this chapter. But first, a look at what's new in BricsCAD V17.

WHAT'S NEW IN BRICSCAD V17

This list of new and changed BricsCAD functions was compiled from version 17.1.07. Bricsys continually updates this software, and so for information on functions added since this book was published, please see http://www.bricsys.com/common/releasenotes.jsp.

Changes are highlighted throughout this book, but be aware that information on theses pages is not comprehensive. Command and variable names new since the last edition of this book are shown in boldface **blue**, updated ones are in boldface **black**. Commands and variables are listed in alphabetical order, sorted into the following sections:

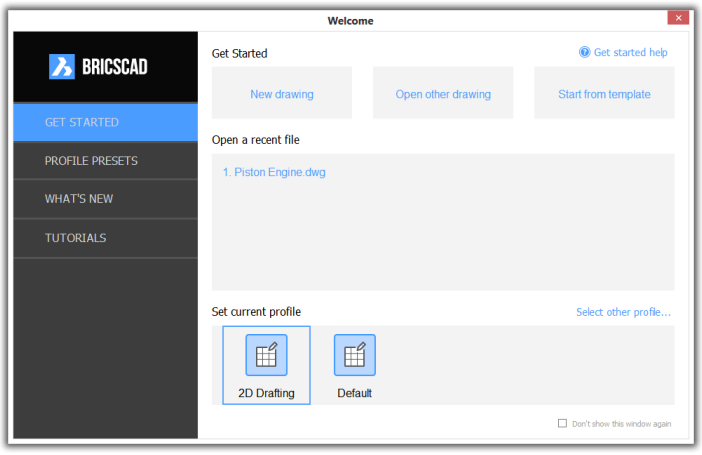
- User Interface
- 2D Drawing and Editing
- Text and Dimensions
- 3D Modeling
- Generated Views
- Rendering
- BIM Module
- Sheet Metal Module
- Communicator Module
- Mapping
- Files
- APIs

BricsCAD V17 installs and runs independently from previous BricsCAD versions.

WHAT'S NEW IN THE USER INTERFACE

BricsCAD V17 supports ultra-high resolution monitors with an extra-large set of icons and reworked dialog boxes.

When BricsCAD V17 starts, it displays a redesigned Getting Started dialog box that makes available the following functions:



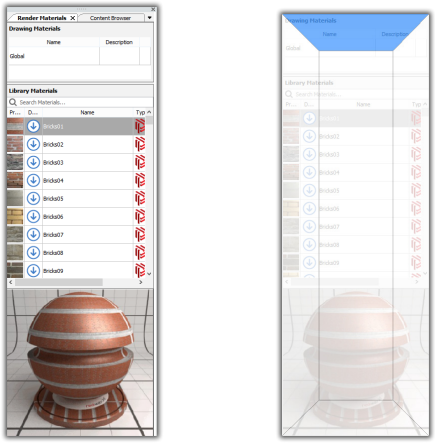
The Welcome window replaces the Getting Started screen

- New and existing drawings
- Workspaces
- User profiles

- Release notes
- Online tutorials

> **TIP** The **OnSwitch** list (commands executed when switching between workspaces) is emptied in V17 so that changes in workspaces are unaffected.

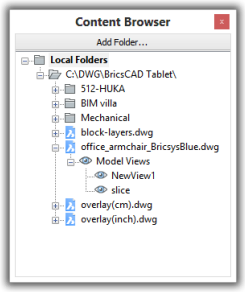
BricsCAD adds more *panels* (new name for 'panes'), and multiple panels can be docked in an overlapping manner. Each docked panel has its own tab: Command Bar, Compositions (for BIM), Content Browser, Layers, Mechanical Browser, Properties Bar, Render Materials, Ribbon, Sheet Sets, Structure, and Tool Palettes.



Left: *Tabbed panels stacked one over the other;* ***right:*** *moving a panel onto others*

- To dock a panel, drag it by its title bar over another panel, choosing one of the five offered locations (see figure above, at right)
- To dock the panel beside, above or under an existing panel, drop it on the corresponding drop location

ContentBrowserOpen command displays the new Content Browser panel with a tree view of drawings in user-selected folders, along with model space views, which can be dragged into the current drawing.



Content Browser panel displaying the content of user-chosen folders

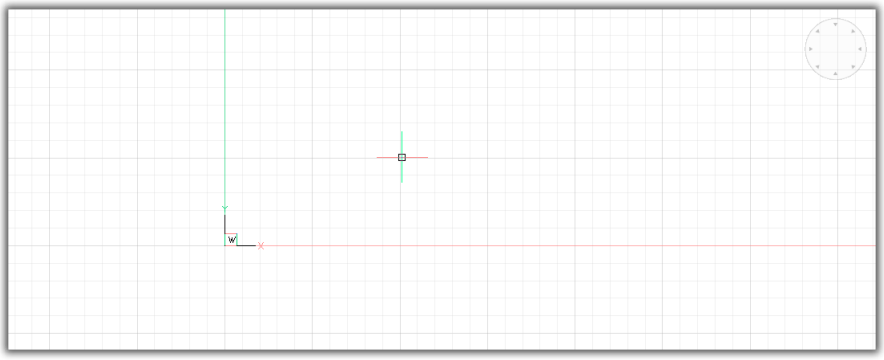
> **TIP** Dragging model views from the Content Browser activates the new **Placeview** command automatically. Bricsys plans to add more drawing content, such as blocks and dimension styles.

ContentBrowserClose command closes the Content Browser pane.

Dynamic UCS behavior on curved surfaces is improved in V17.

Explorer command adds an option to hide and show xref symbols, which applies to all symbol tables. It gains Copy/Paste/Cut for MLeader Styles, MLine Styles, and Layer Filters.

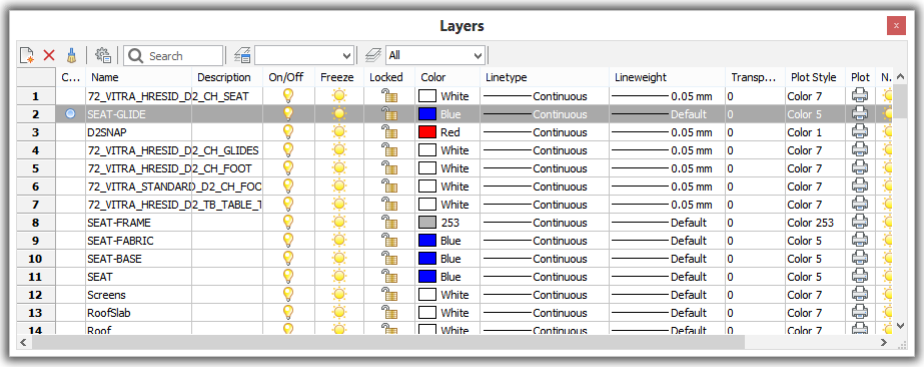
Grid command now draws grid lines partially transparent, and adaptive grid density is reduced to make the grid display less intrusive.



Pale grid lines

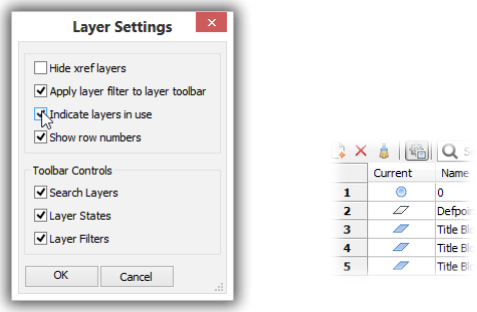
Layer command now displays which layers are in use in the Current column. (You turn it on in the LayerPanelOpen panel.) This command now displays a warning dialog box when opening drawings with more than 250 layer filters, and then offers to remove the filters, as they can cause performance penalties. In any case, the performance of opening drawings with many layer filters is improved.

LayersPanelOpen command displays layer names and properties in a new dockable panel.



New Layers panel

The **Settings** button toggles the display of the panel's UI elements, including the new **Indicate Layers in Use** option.



***Left:** The Layer Settings dialog box; **right:** new layer-in-use icons*

The white layer icon indicates layer is "unused," and so has no entities on it; the blue layer icon indicates layer is being used. (The blue dot indicates the current layer, as in earlier releases.)

LayersPanelClose command closes the dockable Layers panel.

Localization is improved localization for Hebrew menu files.

-Pan command returns the old manner in which pan operations were carried out; **-P** is changed to be its alias.

```
: -PAN
Set pan base point or [Left/Right/Up/Down/PaGe Left/PaGe Right/PaGe Up/PaGe Down]:
Pan displacement point:
```

PromptOptionFormat variable determines how command options are displayed on the command line and in the prompt menu; option 4 is meant for international versions of the software:

PromptOptionFormat	Meaning
0 (default)	Show description only Set end of arc or [**draw Lines**/Angle/CEnter/CLose/...
1	Show keywords only Set end of arc or [**Line**/Angle/CEnter/CLose/...
2	Show description, with keywords in brackets Set end of arc or [Draw lines**(Line)**/Angle/Center**(CEnter)**/Close**(CLose)**/...
3	Show description, with shortcuts in brackets Set end of arc or [Draw lines**(L)**/Angle/Center**(CE)**/Close**(CL)**/...
4	Show local keyword, with global keyword in brackets

TIP Use this variable to keep the command line as compact as possible, or as clear as possible at the expense of extra length.

PromptOptionTranslateKeywords variable toggles the use of international commands. When off, the underscore (_) prefix is not needed during command input; default = on.

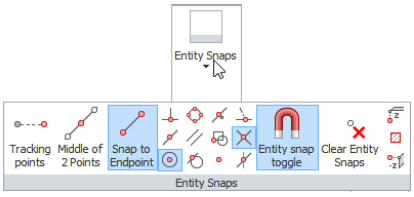
Properties command adds geometry properties for surfaces, such as Watertight, Loops, Holes, Lumps, and Faces. Read-only properties can now be copied to the Clipboard. Iterating through vertices of 2D and 3D polylines is improved.

QuadDisplay variable's value is now preserved when using **F12** to toggle its state.

QuadExpandTabDelay variable specifies the number of milliseconds between hovering over quad tabs and the tabs expanding; default = 50msec.

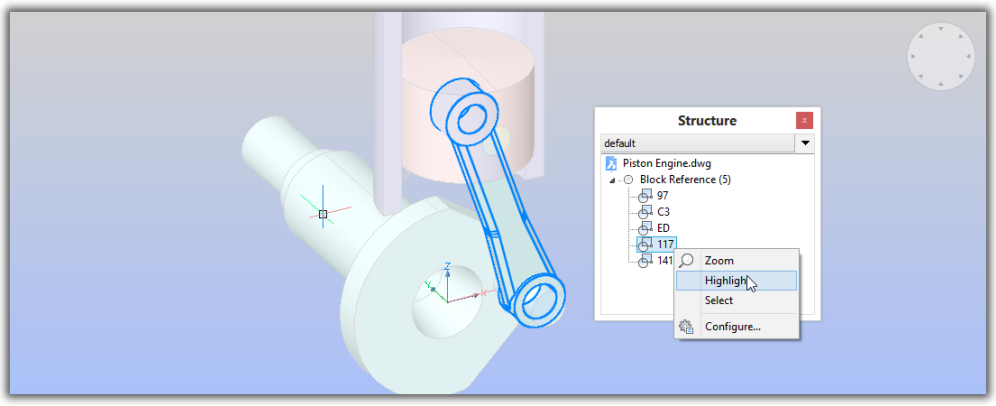
-Rename command now renames layers, blocks, and so on using wild cards.

Ribbon command shows extra items after a panel break in a slide-out panel



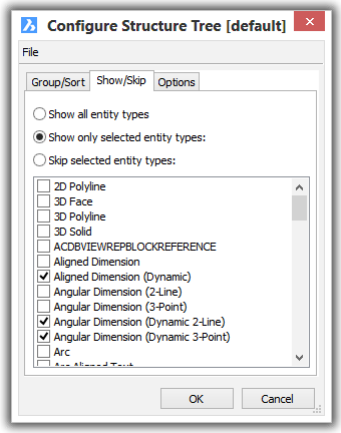
Condensed Entity Snaps panel showing all items in a slide-outpanel

Structure panel displays a structured tree view of the drawing's content. When entities are selected in the structure tree, they are highlighted in the drawing — and vice versa. The panel operates in model space only.



New Structure panel showing the structure of a 3D assembly and highlighting a selected part

The format of the panel can be customized through the Configure dialog box, and then saved and loaded through *.cst* configuration files.



Customzing the display of the Structure panel

StructureTreeConfig command loads *.cst* customize structure files from folders, such as *C:\Users\userid\AppData\Roaming\Bricsys\BricsCAD\V17x64\en_US\Support*.

```
: STRUCTURETREECONFIG
New value for StructureTreeConfig, or . for none/<"bim.cst">
```

ToolPalettes command now displays Group names in the context menu.

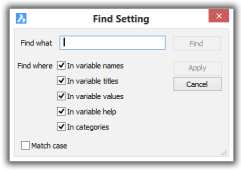
-ToolPanel command opens panels by name at the command bar:

```
: -TOOLPANEL
Enter Tool Panel name <* for all>:
Enter an option [Show/Hide/Toggle] <Show>:
```

TpNavigate command displays the tool palette or palette group specified by the user; meant for use at the command line.

```
: TPNAVIGATE
Specify tool palette to display or [palette Group]:
```

Settings command extends search options to string values in control labels.



Updated Find Setting dialog box

VisualStyle command switches between rendered visual styles quicker.

WHAT'S NEW IN 2D DRAWING AND EDITING

Note: BricsCAD does not have a block editor; the presence of these variables indicates it may be added to a future release.

BlockEditLock variable prevents the Block Editor from opening when a block is double-clicked; default = 0.

BlockEditor variable reports whether the Block Editor is open (read-only); default = 0.

BvMode variable toggles the display of hidden entities in the Block Editor; default = 0.

ResetBlock command resets dynamic blocks to their default values.

Center command and **CENter** entity snap now snaps to circular and elliptical viewports in paperspace.

Fillet command now applies a radius of 0 when the **Shift** is held down while selecting the second entity.

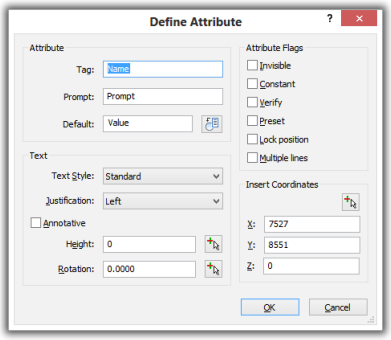
GCE (Geometric Center Esnap) now snaps to the center of rectangular and polygonal paperspace viewports, both clipped and unclipped.

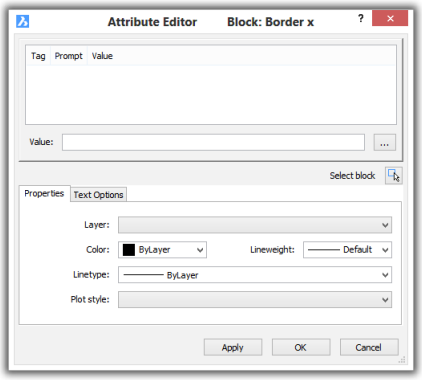
SplinEdit command edits splines:

```
: SPLINEDIT
Select spline:
Edit spline [Close/Join/Fit data/Edit vertex/convert to Polyline/Reverse/Undo/eXit] <eXit>:
```

WHAT'S NEW IN TEXT

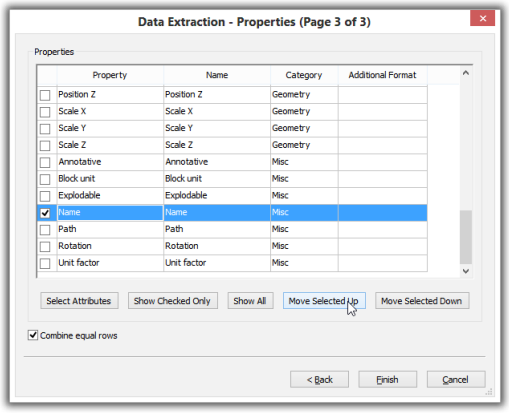
AttDef, **EAttEdit**, and **BAttMan** commands boast improved dialog boxes.





Redesigned attribute dialog boxes

DataExtraction command now moves property rows using the new **Move Selected Up/Down** buttons or with the right mouse button. It now supports properties specific to BIM and sheet metal objects.



Data Extraction dialog box's new Move Selected buttons

Properties command adds the **Misc > Multiple Lines** option to convert single-line attribute definitions to multi-line ones.

Spell command offers improvements and bug fixes.

-Style and **Explorer** commands now show local font name if available, such as Chinese.

Table command now selects a delimiter for *.csv* files from a combo box: semicolon, tab, comma, and space.

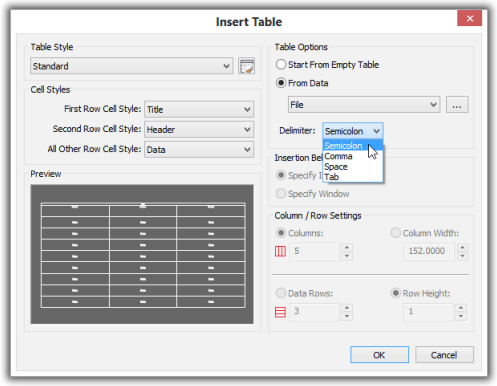


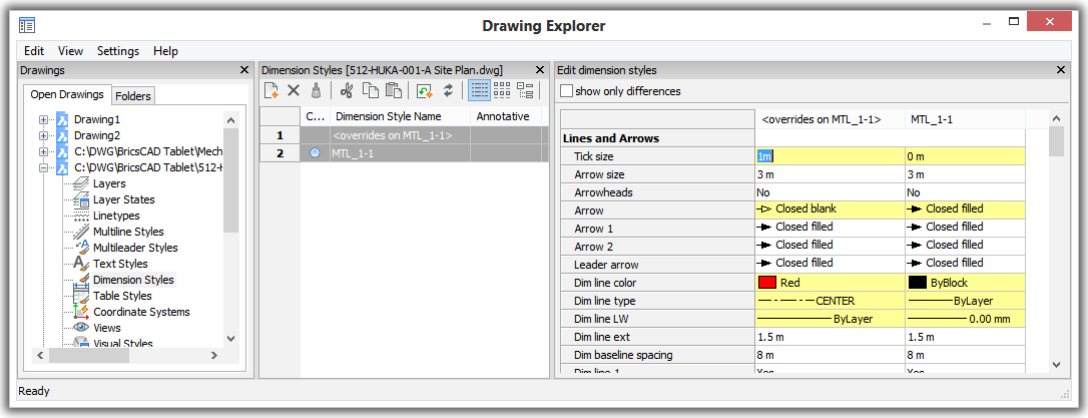
Table dialog box's new Delimiter option

WHAT'S NEW IN DIMENSIONS

aiDimPrec command sets the precision (number of decimal places or accuracy of fractions) of dimension text, and then stores the result in the **dimDec** variable.

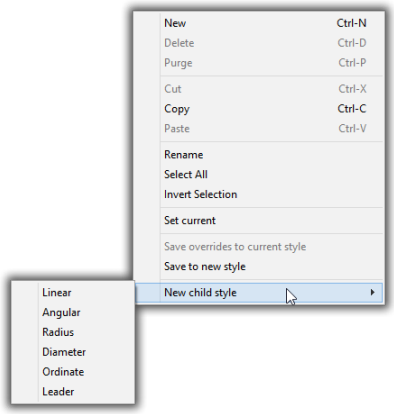
aiDimFlipArrow command mirrors selected arrows of dimension lines.

dimStyle command now offers an **Edit Dimension Styles** panel in the Dimension Styles explorer to see the differences between selected dimension styles (highlighted in yellow); values can be edited directly in the comparison table.



Differences in dimension styles being highlighted in yellow

Dimension style families start with a parent style (ie, a regular dimension style), and the define one or more child styles that are variations of linear, angular, diameter, radius, ordinate, or leader styles.



Creating new child styles

TIP To create a child style, right-click the parent style in the Dimension Styles explorer and then select the **New child style** option in the popup-menu.

Sub-units factor sets the number of sub units to a unit, and is used when the distance is less than one unit. For example, enter 100 if the suffix is m and the sub-unit suffix is to display in cm. This turns 0.96m into 96cm. Dimension styles display of dimension distances less than one unit in sub-units when the dimZIN variable is turned on (normally suppresses leading zeroes).

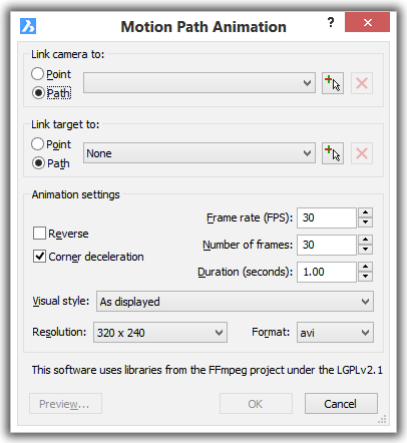
dimTxtDirection variable is added to the Properties panel and the Drawing Explorer.

dimTEdit command now immediately accepts preselected entities, when there is only one entity in the preselection. If more than one, or none, in the preselection, the command asks to select an entity.

WHAT'S NEW IN 3D MODELING

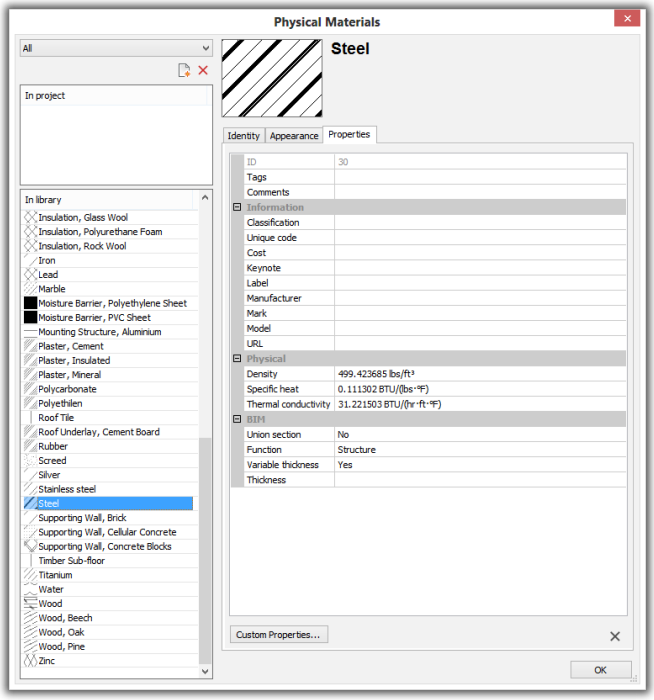
(bm = bricsCAD modeling; dm = direct modeling)

AniPath command records a series of images from a camera moving along a path in 3D models, and then saves it to *.avi*, *.mpg*, or *.wmv* files.



Dialog box for controlling the creation and output of movie files

Component materials define materials with physical properties, so that the mass and other geometric properties are accurately calculated. Materials are accessed from built-in or user-defined libraries of the Physical Materials dialog box. The default material is <Inheret>.



Dialog box for defining real-world properties of materials

TIP There is no command to access this dialog box. Instead, open the BIM Composition panel (right-click a toolbar or the ribbon, and then choose **BIM Composition**). In its toolbar, click the **Physical Materials** button.

---

bmAutoUpdate variable controls whether locally stored copies of external components are reloaded automatically on opening the assembly document.

bmBom command's bill of materials table can now include the **Material** column.

bmBrowser command's Mechanical Browser pane now shows constraint arguments in the tree with their properties. Arguments can be removed from rigid sets or added to them from the current selection set.

bmExternalize command now preserves features and constraints attached to components in a larger number of cases, and now resolves file name conflicts when using the 'automatic mode' option.

bmInsert command now supports the insertion of local components, sheet metal form features, and arranges inserted items in linear arrays. The new Edit option modifies component parameters during insertion.

> **TIP** When a component definition file contains 3D solids on BC_UNITE or BC_SUBTRACT layers, then the solids are added or subtracted from target 3D solid upon component insertion. The Insert as property of the component definition specifies whether the component is inserted as local or external.

bmLocalize command now preserves features and constraints attached to components in a larger number of cases.

bmMassProp command now takes into account the density of materials assigned to components. The group "Density" is removed from component properties and is not taken into account by this command. To define density, materials are to be used instead.

bmReplace command's new **Similar inserts** option control if all similar inserts are replaced or only a selected one. It also preserves features and constraints attached to components in a larger number of cases.

bmUpdateMode variable determines when external assembly components are reloaded:

bmUpdateMode	Meaning
0 (default)	Update only modified components
1	Update all components

dmAngle3d command applies a constraint that controls the top angle of a cone; specify the angle between the axis and the cone face (= cone half angle).

```
: DMANGLE3D
Select first entity or specify [cone Angle constraint]:
Select second entity:
Specify angle value or set [set Axis] <90.00>:
```

dmAudit command replaces the **dmRepair** command to validate 3D solid and 3D surface geometry. The former **dmRepair** command structure and the clarity of reported issues has been improved.

```
: DMAUDIT
Select entities to audit [Entire model] <Entire model>:
Entire model will be processed, number of entities: 2
Choose action [Check/Fix/Options] <Fix>:
Selected count: 2
---------- Solid ------------------------------------------------
    Handle: 393
    Name in Mechanical Browser: Body_1
    Errors: None
---------- Skipped: ---------------------------------------------
       1 Line
No errors were found.
```

Delete key deletes 3D solid sub-entities. **dmDelete** command is replaced by the Delete key, but still works in V17.

dmDistance3d command measures between the nearest points on boundaries, central points, or the axes of geometry on cylinders, circles, and spheres.

```
: DMDISTANCE3D
Select a first entity or specify [Measurement mode]:
Select a second entity or specify [Measurement mode]:
Specify distance value <12.51>:
```

dmExtrudeMode variable controls Boolean operations for the **Auto** option of the **dmExtrude** command:

dmExtrudeMode	Meaning
0	Unite with new 3D solid
1	Create new 3D solids that extrude from a face
2	Subtract from solids that intersect
3 (default)	Both

dmSelect command is enhanced with the following options:

```
: DMSELECT
Select [Selection/Attribute/Relation/Primitive/feaTure/Finish] <Finish>:
Select [Face/Edge/Loop/edge Network] <Face>:
```

- New **Primitive > EdgeNetwork** option to find adjacent edges with similar convexity
- New **Primitive > Loop** option is enhanced to select borders of selected sets of faces
- Selects G1- or G2-connected faces

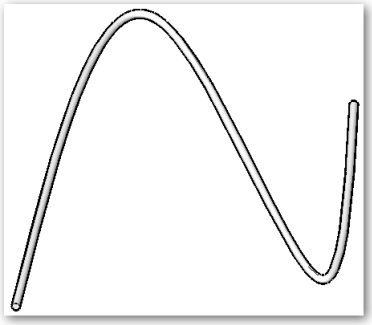
dmMove command now rotates adjacent planar faces, instead of translating edges.

dmRecognize command is no longer affected by design intent options during parametric components recalculations.

dmStitch command no longer preserves tolerance between command runs.

dmThicken command now creates tube-like 3D solids from wireframe entities, such as lines, splines, and polylines.

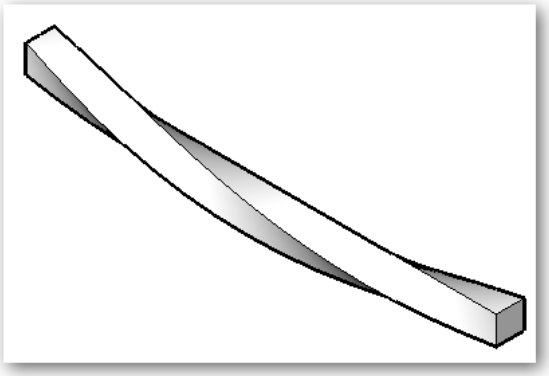
```
: DMTHICKEN
Select entities/subentities to thicken:
Entities in set: 1
Select entities/subentities to thicken:
Specify thickness value:
```



Tube made from a spline with the dmThicken command

dmTwist command modifies 3D solids, 3D surfaces, or 2D regions by twisting them about an axis by a specified angle.

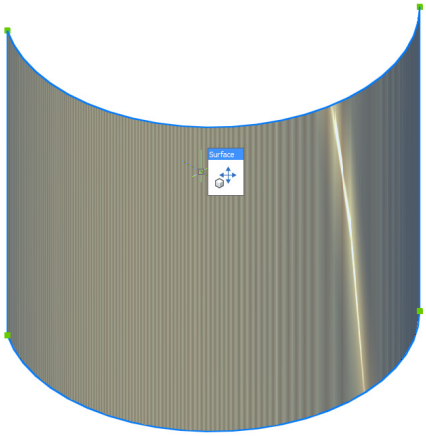
```
: DMTWIST
Select object to twist:
Entities in set: 1
Select object to twist:
Enter start point of twisting axis:
Enter end point of twisting axis:
Pick start point of twisting:
Specify twist angle or set [Continuity]:
```



Square bar twisted by the dmTwist command

Erase command now accepts edges and faces of 3D solids and 3D surfaces, such as to erase a hole in a surface (after all edges are selected).

Extrude command now creates surfaces from open curves, instead of just solids from closed ones.



3D surface extruded from an arc

Interfere command now supports block references and nested selection of 3D solids inside blocks.

Intersect command now combines 3D solids with 3D surfaces or 2D regions.

Loft command now creates surfaces from open and closed curves.

MassPropAccuracy variable defines the number of decimal places, but is now stored as an integer with range of 2 - 12, instead of as a double-precision value (range 0.01-0.000000000001).

MassUnits variable specifies the units the Properties pane uses to report the mass of 3D solids; default = "z lbs stone mg g kg tonne".

PolySolid command can now snap to itself during creation, and now closes itself when the last point equal to the starting point.

```
: POLYSOLID
Current settings: Height = 80, Width = 5, Justification = Center, Separate solids =
On, Dynamic = On
Start point or [Height/Width/Justification/Entity/Separate solids/Dynamic] <Entity>:
Set next point or [draw Arcs/Distance/Follow]:
Set next point or [draw Arcs/Distance/Follow/Undo]:
Set next point or [draw Arcs/Close/Distance/Follow/Undo]:cl
Height of polysolid <80>:
```

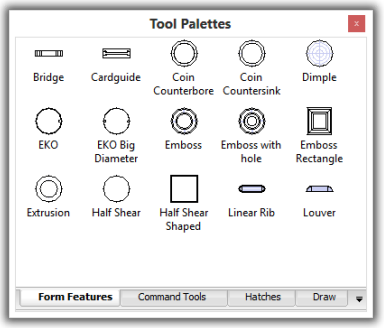
Perspective command now interprets perspective view parameters in a DWG-compatible manner, which may cause perspective views created with older BricsCAD versions to look different when opened in V17.

Properties command now controls the visibility of a particular component insert parameter by the new **Exposed** property.

Subtract command now subtracts 3D solids with 3D surfaces or 2D regions.

Sweep and **Revolve** commands now create surfaces from open curves, instead of just closed ones.

ToolPalettes command now supports components insertion.



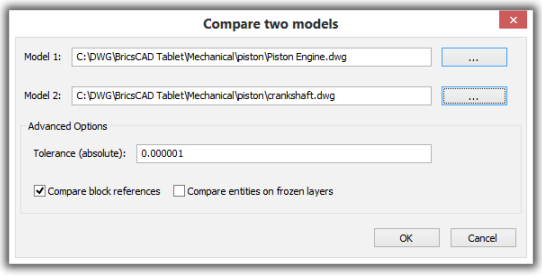
Tool Palettes panel showing available form features

TIP Use the right-click menu to add components from files listed in the Folders tab of the Drawing Explorer to the current tool palette.

XEdges command creates line, circle, or arc entities from the edges of 3D solids, 3D surfaces, and 2D regions.

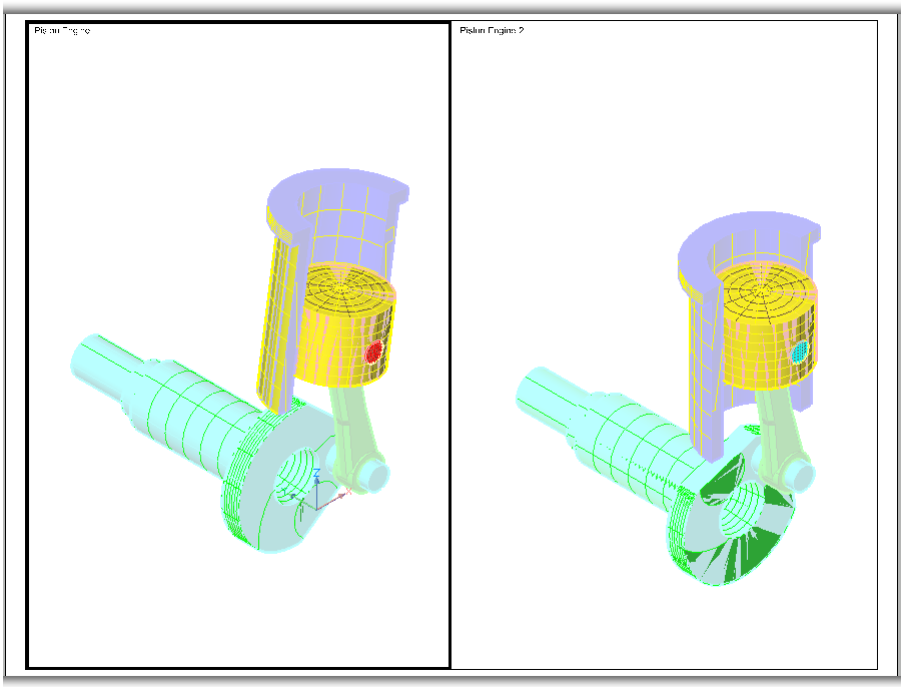
ZINTersection command (or **zint**) is a new 3D snap that snaps to the intersections of edges and tracking lines with faces.

3dCompare command loads two drawing files and then finds differences among 3D solids and surfaces using color coding. New panel in the ribbon.



Selecting two drawings for comparing changes in 3D objects

3dCompareMode variable determines if the results of the comparison results are shown in one or in two viewports; default = 3.



Two viewports showing differences in 3D models

3dCompareMode	Meaning
0	Show models without differences
1	Show differences in layout 'Comparison' left viewport
2	Show differences in layout 'Comparison' right viewport
4	Show differences in model space

3D constraints now take lines, circles, arcs, xlines, and rays as arguments of 3D constraints. If a constraint of the same type already exists, the new constraint is created with the "Disabled" flag.

WHAT'S NEW IN GENERATED VIEWS

AutoVpFitting variable controls the auto-fitting and resizing of viewport borders surrounding generated drawings. It moves derived views automatically when the parent view moves.

AutoVpFitting	Meaning
0	Keeps viewport border sizes fixed
1 (default)	Resizes viewport borders automatically

FlatShot and **SectionPlaneToBlock** commands lose the combo box listing preset orientations; use Dynamic UCS instead to control the orientation during insertion.

GenerateAssocViews variable determines whether the **ViewBase**, **ViewSection**, **ViewDetail**, and **bimSection** commands update the views and associative dimensions attached to 2D drawings automatically when the source 3D model changes. Upon changes to the 3D model, these views will be updated automatically or in course of **bimSectionUpdate** and **ViewUpdate** commands.

GenerateAssocViews	Meaning
0 (default)	bimSectionUpdate and Viewupdate manually update views
1	Automatically updates views and associative dimensions

PlaceView command places a model view from a source drawing into the paper space layout of the current drawing:

- Source drawing is inserted as an xref in the model space of the current drawing, using the same layer(s) as the source drawing
- Paper space viewport is added that matches the source view
- Only layers of the xref are visible in the viewport; view is not disturbed by other drawing content
- When the current drawing belongs to a sheetset, a matching sheetset view is created and a view label block is added

Properties command now shows additional properties when a generated view is selected.

ViewBase command's new **Select objects** option includes or excludes entities from the selection set of the base view.

```
: VIEWBASE
Preset: "None",  View scale: "Adapt to paper size"
Select objects or [Entire model/preseTs] <Entire model>:
```

ViewDetail command has new options:

```
Select option [Scale/Hidden lines/Tangent lines/anChor/Annotation/Boundary/model
Edge] <Cancel>:
```

- **Boundary** option chooses between rectangular or circular boundaries for detail views

```
Detail boundary [Rectangular/Circular]:
```

- **Model Edge** option determines how the leader line is drawn between the detail view and the detail boundary.

```
Model edge type [smooth with Border/smooth with Connection line]:
```

- View properties can be edited before the command is completed

ViewEdit command has new options:

- **Anchor** option fixes view center in paper space
- **Depth** option specifies the depth of sectioned views
- **Select** option includes and exclude objects from base view
- Hidden lines settings and scales are propagated from the parent view to section and detail views.

```
: VIEWEDIT
Select option [Scale/Hidden lines/Tangent lines/anChor/Annotation/Boundary/model
Edge] <Cancel>:
```

ViewSection command's new **Aligned** option chooses alternative projection types.

```
: VIEWSECTION
Select drawing view:
Specify start point of section line or [Type] <Type>:t
Select type [Full/Half/Offset/Aligned] <Full>:


\
```

View properties can be edited before the command is completed.

```
Select option [Scale/Hidden lines/Tangent lines/anChor/Annotation/Depth/Projection]
<Cancel>:
```

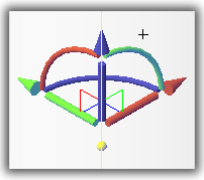
It now uses material-defined hatch patterns for mechanical components and assemblies.

WHAT'S NEW IN RENDERING

MaterialMap command adjusts how rendering textures are mapped on to basic shapes likes planes, boxes, cylinders, and spheres.

```
: MATERIALMAP
Select an option [Box/Planar/Spherical/Cylindrical/copY mapping to/Reset mapping]
<Box>:c
Select faces or entities:
Entities in set: 1
Select faces or entities:
Edit the mapping or [reseT/sWitch mapping mode] <Accept current mapping>:
```

The manipulator gizmo controls the origin, rotation angle, and scale factor of the texture.



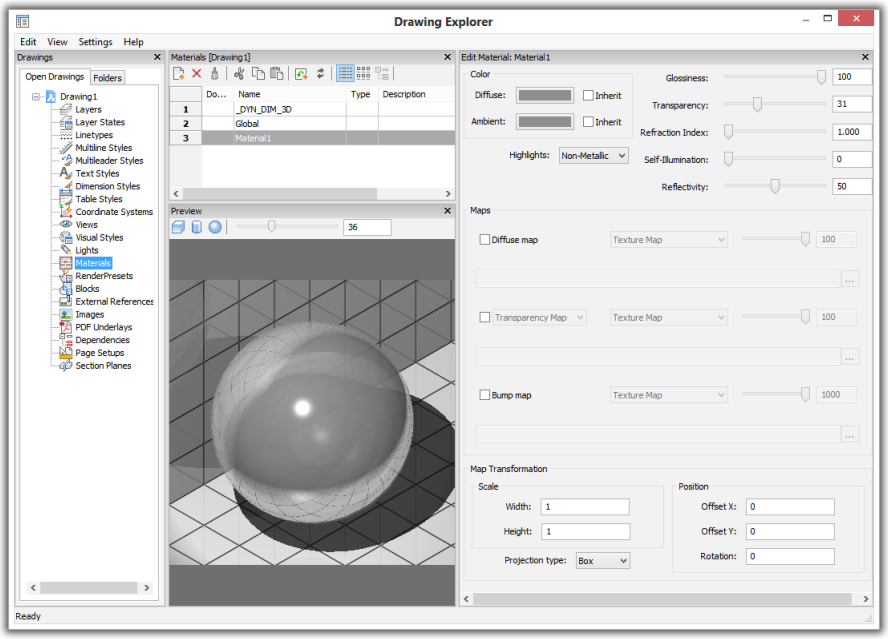
3D gizmo for controlling position of materials on surfaces

MatchPerspective command changes the perspective viewpoint of the current view in model space view to match a background image; this is done by selecting at least three point pairs. The command works only when the Perspective variable = on.

```
: MATCHPERSPECTIVE
Enter Model Point:
Enter Image Point or [Undo]:
Enter Model Point or [Undo]:
Enter Image Point or [Undo]:
Enter Model Point or [Undo]:
Enter Image Point or [Undo]:
Enter Model Point or [Undo] <Match>:
```

Materials commands adds the following functions:

- New columns indicate the render material definition type — regular or RedWay — and the download status
- New option convert RedWay material definitions to a regular definitions
- New preview object size control
- Each material now has a projection type: planar, box, cylinder, or sphere



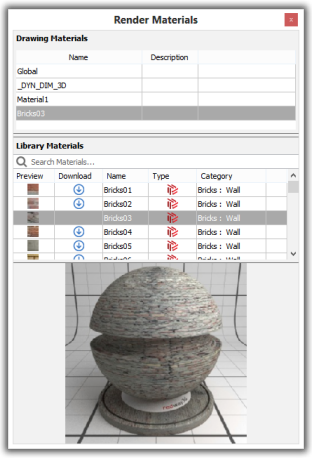
Renovated Materials section in the Drawing Explorer

- **Width** and **Height** scale values are interpreted differently, depending on the projection type:

Projection Type	Width	Height
Cylindrical	Number of cylinder rounds for full texture width	Number of drawing units correspond to full image height
Spherical	Number of sphere rounds for full texture image width	Number of sphere rounds covered by one full image height

TIP Double-click the material's download icon to start downloading.

MatBrowserOpen command (opens the Render Materials pane) now supports double-clicking a material to open it in the Materials Explorer for editing. Materials can be drag-and-dropped from the Library Materials list to the Drawing Materials list.



Updated Render Materials panel

WHAT'S NEW IN THE BIM MODULE

BIM is an optional extra-cost add-on as of V17. (bim = building information modeling)

BricsCAD BIM V17 is certified for IFC export at IFC2x3 Coordination View V2.0 level. BIM elements carry all of the properties defined by the IFC2x3 Coordination View CV2.0, and are accessible in the Properties panel.

bimAutoUpdateRoom variable automatically updates rooms when bounding walls are modified; new walls are not detected.

bimClassify now has the **Other** option to classify the following new BIM elements: Covering, CurtainWall, FlowTerminal, Footing, FurnishingElement, Member, Pile, Railing, Ramp, RampFlight, Roof, Site, Stair, and StairFlight. This command applies Window and Door classifications on window and door definition files.

TIP To reclassify a drawing, enter the **bimClassify** command, select the **Window** or **Door** option, and then press **Enter** to select nothing. The Properties panel edits the drawing's Window and Door properties when nothing in the drawing is selected.

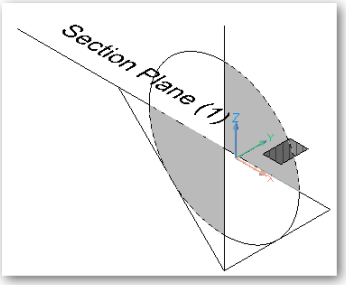
bimRoom command defines a room by clicking inside a room area or by selecting a 3D solid.

```
: BIMROOM
Pick a point or [select 3d Solid]:S
Select 3d solid or [pick a Point]:
Successfully created a room.
```

TIPS Rooms keep the relationship with their bounding walls. When clicking a point inside the room area, BricsCAD uses dynamic UCS to defines the bottom plane of the room. BricsCAD finds the area enclosed by walls, and then places a room marker consisting of a block made of a hatch and attributes for the room's name, number, and area.

When a room is defined by the click method, then it will report the finishing materials of the wall sides inside the room.

bimSection command adds the **Detail** option, which creates a section with Volume state by default.



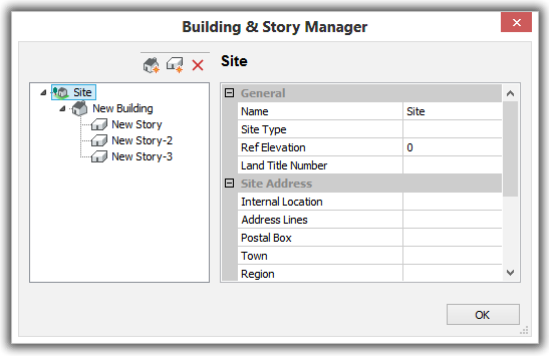
Room section label

TIPS To create the volume section, the command prompts for three points using dynamic UCS to define the base plane of the box and becomes section plane.

Starting **bimSection** from the Quad creates detail section boxes based in the same plane as the section over which the cursor is hovering.

bimSection now displays hatch patterns from compositions live on the 3D model when the Clip Display property is active.

bimSpatialLocations command opens the Buildings & Stories Manager dialog box to create and edit the site (one per drawing), buildings (one or more per site; default = 3), and stories (one or more per building).



Buildings & Stories Manager dialog box

bimUpdateRoom command executes the room-finding algorithm to redefine the room, such as if new walls were added.

DataExtraction command now exports the properties of all BIM elements, including windows, doors, and rooms.

PlaceView command drags 3D views from the Content Browser onto the paper space of a section result drawing.

Properties command is updated for BIM to list **Display Composition**. When on, the selected 3D solid shows its composition of ply faces, which can then be separately selected.

StructureTreeConfig variable loads a .cst "configuration structure" file that formats the new Structure panel. In BIM, it examines all aspects of the BIM model, and can be customized by the user. The default structure organizes the building spatially: first by Building, then by Story, BIM type, and composition. The new Structure panel groups sections by type: Section, Plan, Elevation, or Detail. See comments on the Structure panel earlier in this chapter.

TIP The Structure panel can configured to group and sort by any property, including all IFC properties.

WHAT'S NEW IN THE SHEET METAL MODULE

The Sheet Metal module is an optional extra-cost add-on. (sm = sheet metal)

smBendLineExtentValue variable specifies the bend line properties; default = 0.25.

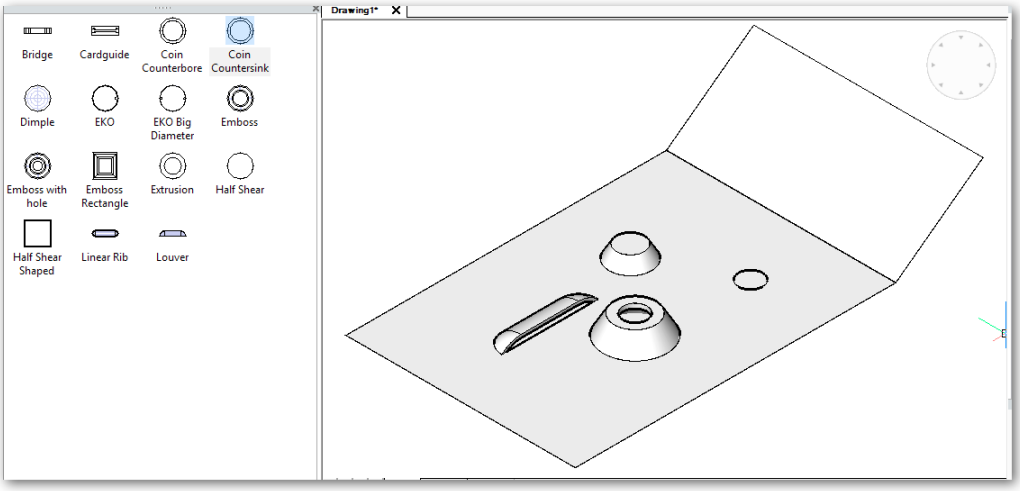
smConvert now recognizes cylindrical lofted bodies as lofted bends, form features in imported geometry, and more types of wrong bend features.

smExportOsm command now creates *.osm* files when bends are adjacent to lofted bends, and adds information about component materials to *.osm* files.

smFlangeBend command bends existing flanges along lines, obeying the k-factor for the given bend radius.

smFlangeEdge command now improves relief creation.

smForm command converts a selected set of faces to form features. A drawing file with a user-defined form feature can be saved and then used with the bmInsert command to insert the form feature.



Adding form features to a sheet metal part

TIP *Form features* are a new kind of sheet metal feature that mimics applying a forming tool to the sheet metal, such as bridges, louver, and embosses. They inserted from built-in or user-defined libraries; BricsCAD recognizes form features in imported geometry. Form features are listed in the Mechanical Browser pane with their parameters; they can be edited directly or parametrically through Properties panel. *C:\Users\userid\AppData\Roaming\Bricsys\BricsCAD\V17x64\en_US\Support\DesignLibrary\SheetMetal\FormFeatures*

smFormFeatureUnfoldMode variable controls the appearance of form features in 2D and 3D unfolded model representations; this variable must be modified through the Settings dialog box.

smFormFeatureUnfoldMode	Meaning
0	Keep
1	Remove
2	Project
3	Contour
4 (default)	Symbol

smJunctionCreate adds the option to select "Entire model" and 3D solids.

smKFactor variable specifies the default K-factor; default =0.27324.

smLoft command's new **Auto** option for fillet radius creates bodies with the smallest possible fillet radius (given the thicken type).

smReliefSwitch command adds options for bend reliefs: "Switch to Smooth", "Switch to Round" and "Switch to Rip". The new auto value for corner relief extensions now means "Keep the extension, which is set in the feature. Switching corner reliefs near flange splits (a mitter) to V-type are automatically converted to two smooth bend reliefs.

smReliefCreate command optionally forces the creation of bend reliefs. It creates correct relief geometry when the bend radius is not equal to the default bend radius set in the drawing.

smRepair command replaces the smRethicken command, which is removed. It now repairs wrong bends by converting them into regular bends, changes the "Enable lofted bend repair" prompt to "Merge lofted bends", with improved support for adjacent lofted bends. It gains automatic repair of coincident faces cases for WrongBend features.

smReplace command replaces form features (including recognized ones) in sheet metal parts with form features from built-in or user library — even if their dimensions are different.

smSelect command selects hard edges, same and similar form features, and is added to the Select section of the Quad menu.

~~smRethicken~~ command is removed; its function is replaced by the smRepair command.

_Sm_Thickness component parameter can be edited during and after insertion with the **bmInsert** command.

smUnfold command adds information about component materials to *.dxf* files, and displays a warning message about torn lofted bends.

WHAT'S NEW IN THE COMMUNICATOR MODULE

Communicator is an optional extra-cost add-on. BricsCAD V17 is not compatible with Communicator V16, and so Communicator must be upgraded.

ExportProductStructure variable determines whether the product (assembly) structures are exported.

ImportHiddenParts variable controls if hidden parts are imported

Communicator now exports the following data:

- Product (assembly) structures to IGES/STEP

Communicator now imports the following data:

- Materials with physical properties, if they are assigned to the parts of imported products
- Root assembly component names are set to the imported assembly file name automatically.
- Alternate search paths search for imported assembly parts from Creo, Inventor, NS, SolidEdge, and Solidworks file.
- XCGM file format

WHAT'S NEW IN MAPPING

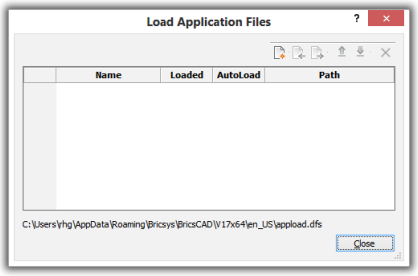
BricsCAD V17 now supports the following coordinate reference systems:

- Czechia/Slovakia S-JTSK
- Croatia EPSG 3765
- Netherlands EPSG 28992

GeographicLocation command gains a fast filter-as-you-type control when searching coordinate reference systems.

WHAT'S NEW IN FILES

AppLoad command's dialog is redesigned to make auto-loading applications easier, adding support for.LSP, .NET, and .VBA (in addition to .BRX and .TX).



Redesigned AppLoad dialog box

CheckDwlPresence variable reports whether *.dwl* and/or *.dwl2* locks files are associated with the drawing being opened.

-eTransmit command is the new command line version of the **eTransmit** command. Both commands now handle the additional files required for BIM and mechanical assembly modeling.

ExportPdf command takes over PDF exporting from the **Export** command. Text in exported PDF files is now searchable for all visual styles. (Text in clipped inserts is not yet exported as searchable text.)

ImageAttach command now allows multiple selections of images from a single folder to attach multiple images at once. This is especially useful for images with geo-information attached. The images are laid on top of each other.

-ImageAttach now support relative and absolute paths in the command line version.

PdfMergeControl variable determines how overlapping lines are printed.

PdfMergeControl	Meaning
0 (default)	Lines overwrite
1	Lines merge

XrefOverride variable controls the display of properties like color, linetype, lineweight, transparency, and plot style in referenced layers.

WHAT'S NEW IN APIs

ACIS is upgraded to v2017 1.0 (R27).

BricsCAD V17 is compiled with Visual Studio 2013 (platform toolset = v120), and so to be compatible C++ extension .dlls need to be compiled with the same platform toolset.

VBA is upgraded v7.1. It add support for the following items:

- Allows 64-bit operations and is compatible with earlier versions
- Provides compilation constants VBA7 and Win64
- Adds keywords LongLong, LongPtr, and PtrSafe

TIP VBA is no longer installed by default, but requires a separate installation from the VBA subfolder of the application installation folder, such as in *C:\Program Files\Bricsys\BricsCAD V17 en_US\VBA*.

The following BricsCAD-specific APIs are added to BRX:

- An API for the Quad in BRX and .Net
- An unmanaged C++ Ribbon API
- An API for 3D constraints and parameters
- Subentities are supported by the C++ OPM API part of BRX
- Ribbon API was added for BRX and .NET

BRX 17 interface is updated to be source code-compatible with ARX 2015/2016 SDK, such as overrules. As there are some exceptions, BRX supports newer and legacy interfaces when possible.

> **TIP** Sample applications installed with BricsCAD are updated to demonstrate these new APIs, such as under the *C:\Program Files\Bricsys\BricsCAD V17 en_US\API* folder.

LICENSE REQUIREMENTS

Pro or Platinum license is required for rendering, 3D modeling, and Drawing Views functions.

Platinum license is required for creation of 3D Constraints, Mechanical Assembly design, and Deformable Modeling functions.

BIM design and Sheet Metal design functions require a separate license for each on top of a BricsCAD Platinum license.

For the latest information about what's new in V17, including bug fixes, see https://www.bricsys.com/common/releasenotes.jsp.

CHAPTER TWO

Comparing User Interfaces

BRICSCAD LOOKS A LOT LIKE AUTOCAD, AS YOU CAN SEE ON THE NEXT PAGE. JUST like AutoCAD, BricsCAD defines aspects of its user interface by several means, such as from the content of CUI files and the settings of variables. While AutoCAD has overall more capability in CUI, BricsCAD provides greater user control through its extensive collection of variables.

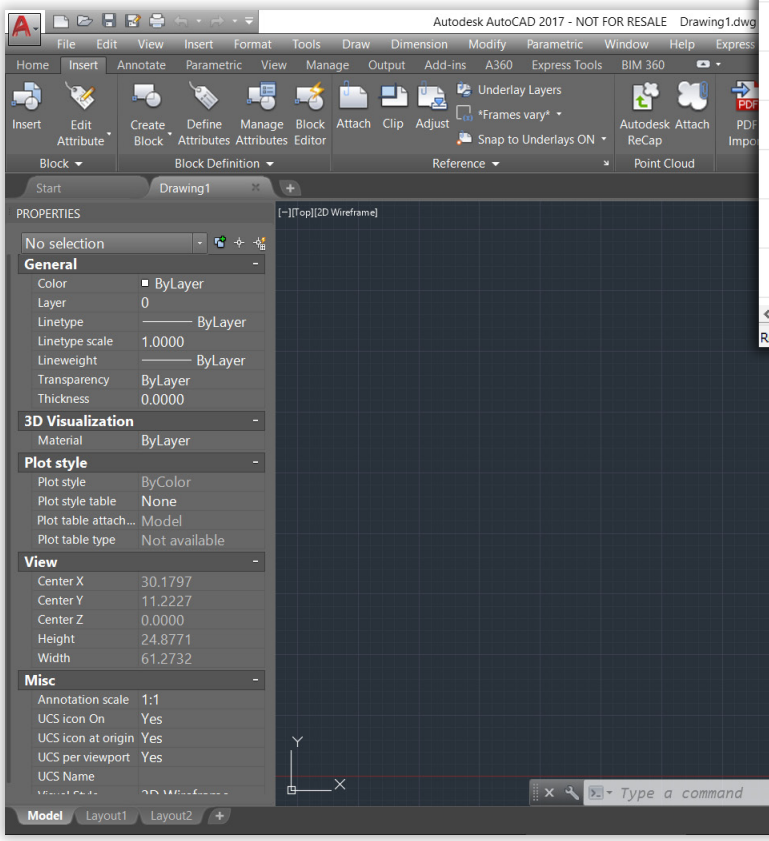
In this chapter, you learn about the similarities (and differences) between the user interfaces of the two CAD systems, specifically in the following areas:

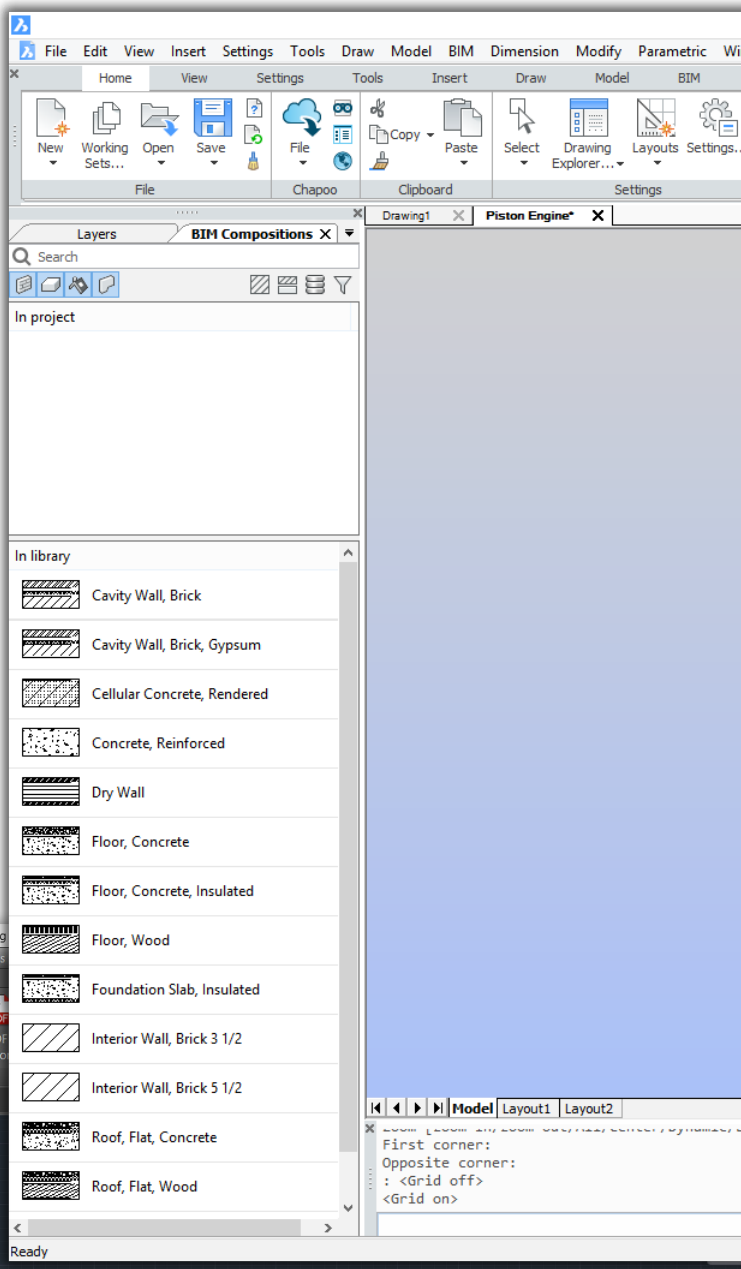
- Start screen
- Command line and prompts
- Prompt menu (BricsCAD only)
- Quad cursor (BricsCAD only)
- Settings (BricsCAD) and Options (AutoCAD) dialog boxes
- Properties, Layer, Tool, and Sheet Set palettes
- Mechanical Browser (BricsCAD) and Parametrics Manager (AutoCAD)
- Status bar
- Selection sets
- Working sets (BricsCAD only)
- Tips Widget (BricsCAD Only)
- Differences in View Cubes
- Drawing Explorer (BricsCAD) and Design Center (AutoCAD)
- Chapoo (BricsCAD) and Autodesk 360

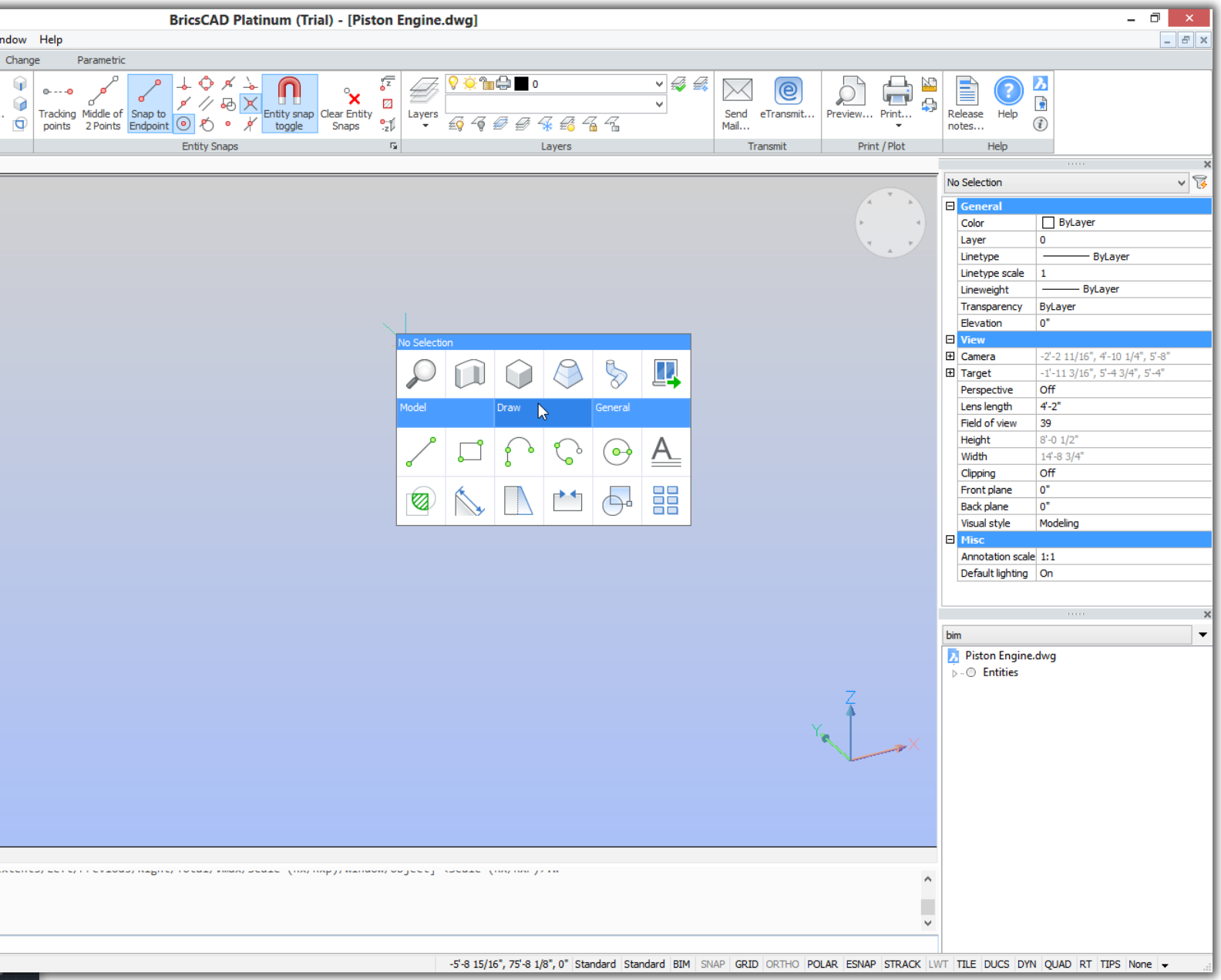
COMPARISON OF USER INTERFACES

Right: *BricsCAD V17 in Mechanical workspace*

Below: *AutoCAD as it appears when first launched in its default dark user interface*






BricsCAD Platinum (Trial) - [Piston Engine.dwg]
ndow Help
Change Parametric
Tracking points
Middle of 2 Points
Snap to Endpoint
Entity snap toggle
Clear Entity Snaps
Entity Snaps
Layers
Layers
Send Mail...
eTransmit...
Transmit
Preview...
Print...
Print / Plot
Release notes...
Help
Help
No Selection
General
Color ByLayer
Layer 0
Linetype ByLayer
Linetype scale 1
Lineweight ByLayer
Transparency ByLayer
Elevation 0"
View
Camera -2'-2 11/16", 4'-10 1/4", 5'-8"
Target -1'-11 3/16", 5'-4 3/4", 5'-4"
Perspective Off
Lens length 4'-2"
Field of view 39
Height 8'-0 1/2"
Width 14'-8 3/4"
Clipping Off
Front plane 0"
Back plane 0"
Visual style Modeling
Misc
Annotation scale 1:1
Default lighting On
No Selection
Model
Draw
General
bim
Piston Engine.dwg
Entities
Z
Y
X
-5'-8 15/16", 75'-8 1/8", 0"
Standard
Standard
BIM
SNAP
GRID
ORTHO
POLAR
ESNAP
STRACK
LWT
TILE
DUCS
DYN
QUAD
RT
TIPS
None

SUMMARY OF USER INTERFACE ELEMENTS

The UI elements discussed in this chapter are shown in **boldface**.

UI Element in AutoCAD	Equivalent Element in BricsCAD
...	**Working (drawing) sets**
Customizable user interface	Customizable user interface
Menu bar (turned off in default workspace)	Menu bar
Toolbars (turned off in default workspace)	Toolbars
Scroll bars	Scroll bars
Tooltips	Tooltips
Layout tabs	Layout tabs
Status bar	Status bar
Workspaces	User Profile Manager
Rollover tooltips	Quad Quick Properties
Drawing tabs	Drawing tabs
Ribbon	Ribbon
QuickView layouts and drawings	...

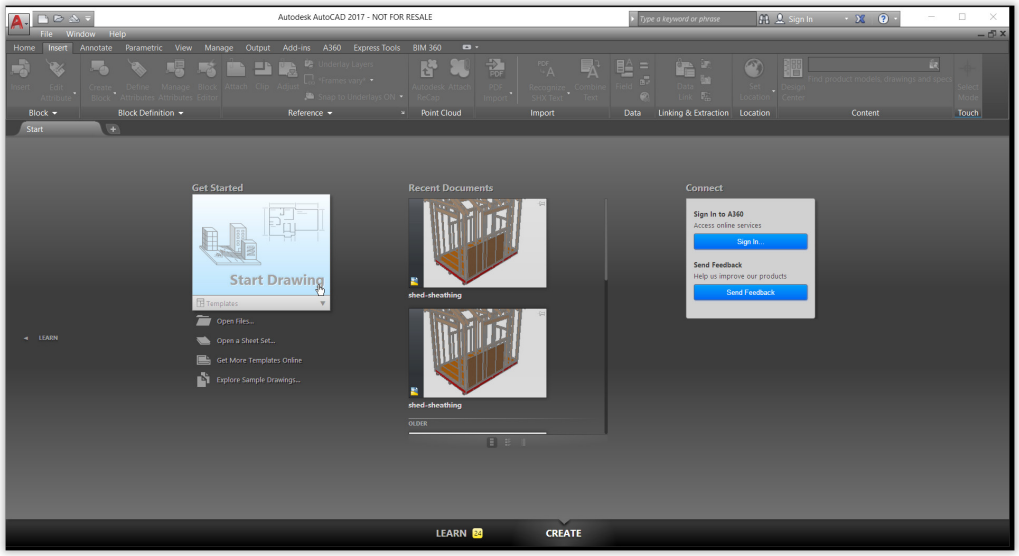
On the Drawing Screen	
...	**Quad cursor**
...	**Tips widget for shortcut keystrokes**
Tri-color cursor	Tri-color cursor
UCS icon & dynamic UCS	UCS icon & dynamic UCS
Aperture & pickbox cursors	Aperture & pickbox cursors
Grips	Grips
Dynamic block grips	Dynamic block grips
Selection highlighting & previews	Selection highlighting & previews
AutoSnap markers & autotrack vectors	AutoSnap markers & autotrack vectors
Selection modes: 14	Selection modes: 18
Subentity selection	Subentity selection
Navigation cube	**Look From widget**
Steering wheels	...

Command Bar and Mouse	
...	**Customizable command prompt**
...	**Prompt (options) menu**
Keyboard input	**Keyboard input** (see Appendices A, B, and C)
AutoComplete	**AutoComplete**
Dynamic input	**Dynamic input**
Keyboard shortcuts	Keyboard shortcuts (see Appendix C)
Double-click actions	Double-click actions (see Appendix D)
Mouse buttons	Mouse buttons (see Appendix D)
3D Mouse	3D Mouse (see Appendix D)
Shortcut menus	Shortcut menus

Information Centers	
...	**Prompts on status bar**
DesignCenter	**Drawing Explorer**
Properties palette	**Properties bar**
Tool palettes	**Tool palettes bar**
Sheet set manager	**Sheet sets**
Parameters manager	**Mechanical browser / Hardware library**
InfoCenter	Help
Quick Properties	Quick Properties
Quick Access toolbar	Quick Access toolbar

Start Screens

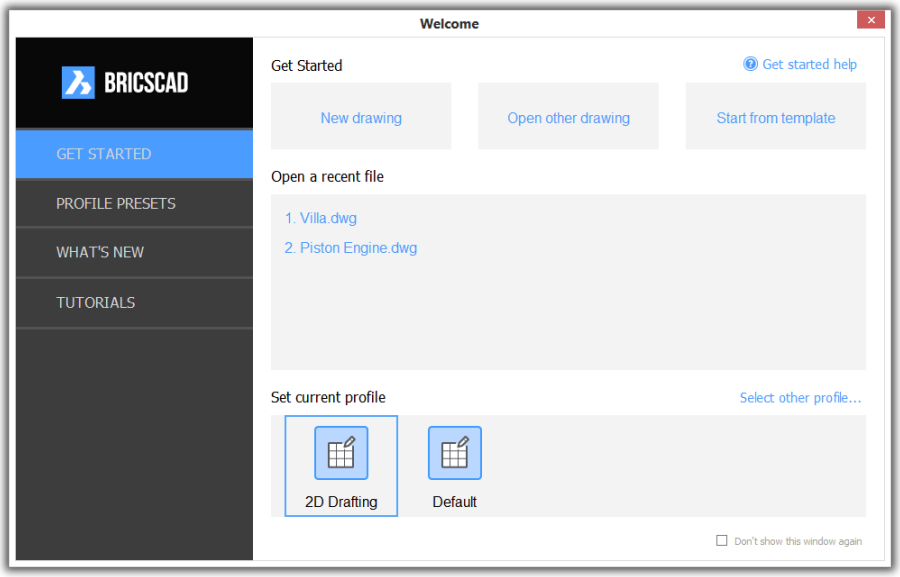
BricsCAD and AutoCAD launch with start screens. AutoCAD's is illustrated below.



One of the pages of the start screen in AutoCAD

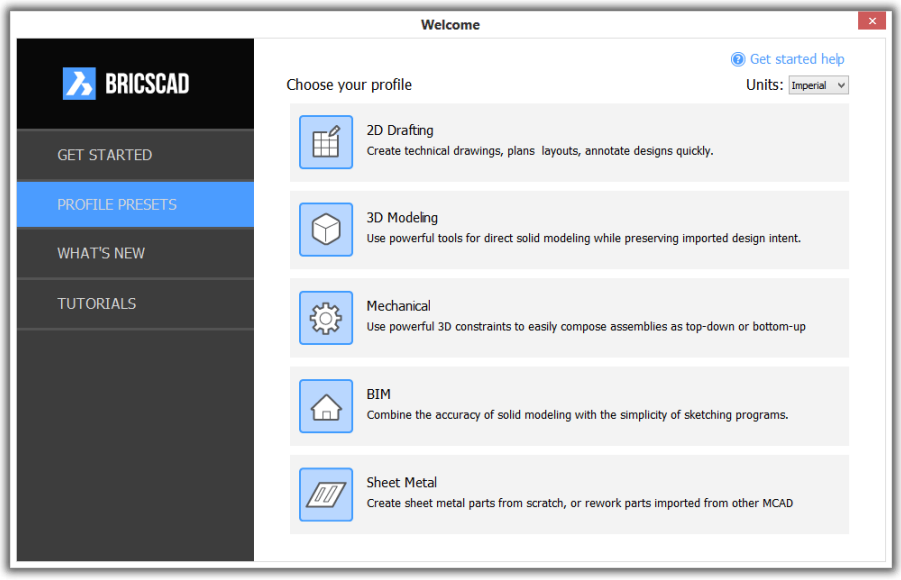
TIP As of AutoCAD 2015, Autodesk made the dark interface the default color scheme for the Windows version of the CAD program. This color scheme places white text on a dark background for many UI elements, although its dialog boxes defy the scheme by continuing to be the other way around: black text on a light background. Screen grabs of AutoCAD in this book reflect the default setting.

The start screen for BricsCAD accesses workspaces, starts new drawings, opens previously-opened drawings, and accesses online tutorial videos. (This screen was redesigned in V17.)



The new start screen in BricsCAD V17

The **Profile Presets** button takes you to profiles and workspaces.

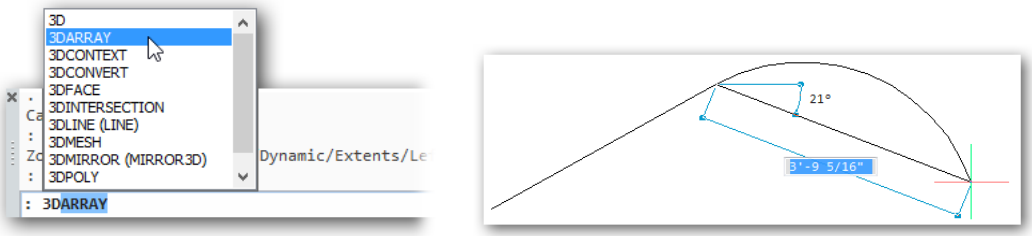


Profile Presets showing workspaces and profile names

The **What's New** button takes you to the list of what is new, improved, and fixed in each release of BricsCAD; you can read the release notes online at https://www.bricsys.com/common/release-notes.jsp. The Tutorials button accesses video tutorials on using BricsCAD, also found at https://www.bricsys.com/tv.

Variations in User Interface

BricsCAD and AutoCAD sport user interfaces that look similar to each other. Both offers ribbons, toolbars, menu bar, and status bar. For command input, both provides autocomplete, dynamic input, palettes, shortcut menus, and so on. The figure below shows BricsCAD with autocomplete (left) and dynamic input (right).

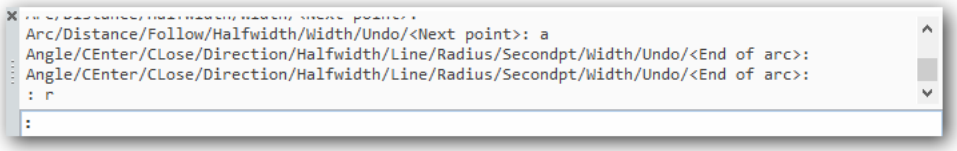


***Left:** BricsCAD command bar with AutoComplete; **right:** Dynamic input in BricsCAD drawing area*

BricsCAD has some user interface differences from AutoCAD in areas such as the command prompt wording, the prompt menu, and some command options. Let's look at these.

':' VS 'TYPE A COMMAND'

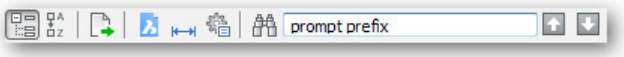
For its command prompt, BricsCAD uses very compact ':' prompt to indicate it is ready for you to enter a command. Old releases of AutoCAD used 'Command:', but newer releases display the even longer 'Type a command'.



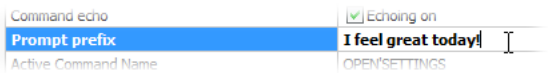
Bricsys command prompt showing a colon (:)

Customizing the Command Prompt (BricsCAD only)

If you prefer to see AutoCAD's prompt wording or anything else in BricsCAD, you are free to change the display. To do so, open the Settings dialog box, like this: enter the **Settings** command, and then in the search field enter 'prompt prefix'.

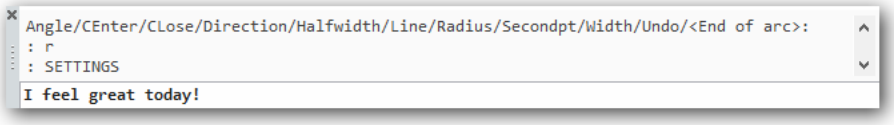


BricsCAD jumps to the Prompt Prefix field, in which you can enter any text you like, even silly things.



Changing the prompt displayed by the command bar in BricsCAD

Exit the dialog box (click big red X), and the new prompt text appears immediately.



BricsCAD command prompt changed

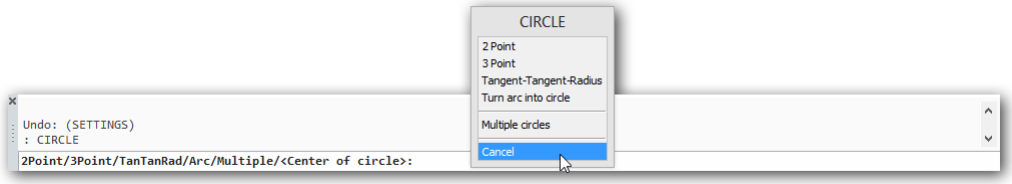
(NEW IN V17) The **PromptOptionFormat** further customizes command prompts by making them more or less verbose. Option 4 is is useful for international versions of the software:

Value	Meaning	Example
0 (default)	Show description only	Set end of arc or [**draw Lines**/Angle/CEnter/CLose/...
1	Show keywords only	Set end of arc or [**Line**/Angle/CEnter/CLose/...
2	Show description, keywords in brackets	Set end of arc or [Draw lines(**Line**)/Angle/Center(CEnter)/...
3	Show description, shortcuts in brackets	Set end of arc or [Draw lines(**L**)/Angle/Center(CE)/Close(CL)/...
4	Show local keyword, global keyword in brackets	

(AutoCAD does not provide customization of the command line wording.)

PROMPT MENU (BRICSCAD ONLY)

One of BricsCAD's user interface elements not found in AutoCAD is the *prompt menu*. This is a floating menu that appears whenever a command has options. The idea behind the prompt menu is to let you operate BricsCAD without a command prompt area; also, it provides a way to choose options with a mouse instead of using the keyboard.



Left: *Command bar in BricsCAD displaying options of the Circle command*; **center:** *Prompt menu displaying equivalent options*

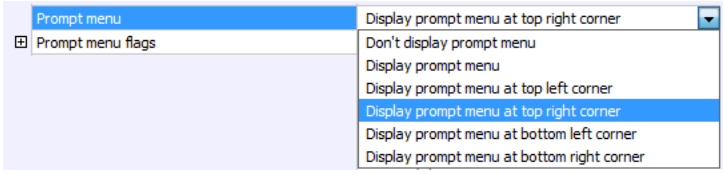
In the figure, you see command line window (at left) and the prompt menu in the center. As the Circle command progresses, the prompts in the command bar and the prompt menu match one another. You are free to specify options through the following inputs:

- At the keyboard type in option abbreviations
- With the mouse choose among options on the prompt menu
- To cancel the command in progress, press **Esc** or click **Cancel**

In some cases, the prompt menu does not appear, such as when BricsCAD prompts you to select objects or when a command displays a dialog box.

Controlling the Prompt Menu (BricsCAD only)

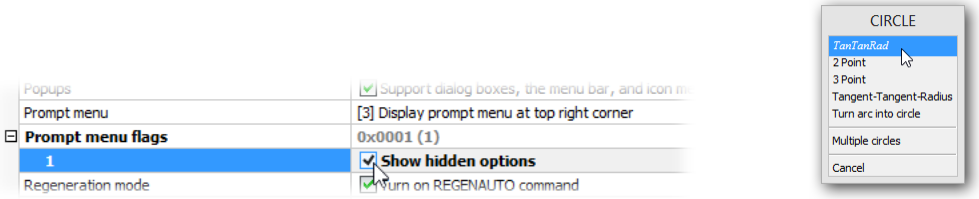
You turn the prompt menu on and off, and specify its location on the screen. In the Settings dialog box, search for 'prompt menu', and then change a setting:



Settings for the prompt menu

The **Don't Display** and **Display** options determine whether the prompt menu is seen. The **Corner** options position the prompt menu towards one of the four corners of the drawing area. Or, you can just drag the menu to any convenient location, such as to a second monitor. BricsCAD remembers the location.

The **Prompt Menu Flags** option is a bonus that forces the prompt menu to display hidden option names. These bonus options are shown in italic text, such as *TanTanRad* in the figure below.



Left: *Toggling hidden prompt menu items;* **right:** *Hidden items, such as TanTanRad, as displayed in italics*

Additional Command Options (BricsCAD only)

You may notice that BricsCAD's Circle command contains more prompts than does AutoCAD. It is is not uncommon for BricsCAD to offer drafters useful commands, options, and variables that are not available in AutoCAD. The following table compares the Circle command's prompts of both programs:

AutoCAD Option Wording	BricsCAD Option Wording	Notes
Specify center point	Center of circle	Default option for both CAD programs
2P	2Point	
3P	3Point	
Ttr	TanTanRad	
...	Arc	Converts arcs into circles (not in AutoCAD)
...	Multiple	Draws multiple circles (not in AutoCAD)

AutoCAD compensates for the options missing from its Circle command through additional commands. To convert an arc into a circle, use the Join command (also found in BricsCAD). To draw multiple circles during one command, use the Multiple modifier (also found in BricsCAD).

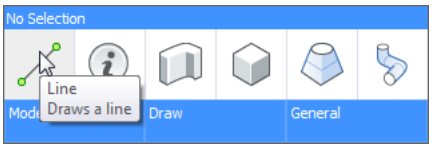
THE QUAD (BRICSCAD ONLY)

The Quad incorporates drawing, editing, and information commands into a single cursor. The multifunction cursor takes its cue from the "heads-up" style of computer interface design, placing at the drawing area many useful commands. It is unique to BricsCAD; AutoCAD does not have this kind of an interface.

The Quad is normally not visible; most of the time you see the usual tri-color cross hair cursor. You access it for drawing, properties, and editing like this:

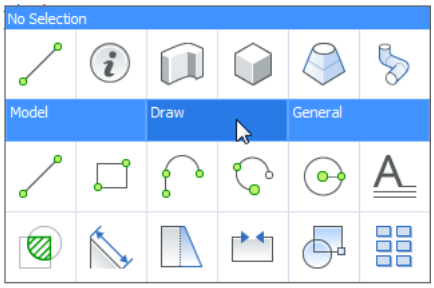
Drawing with the Quad

When you right-click an empty part of the drawing, BricsCAD displays the Quad with drawing and inquiry commands.



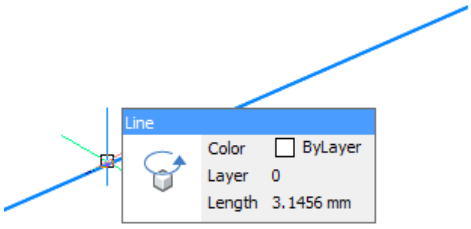
If the Quad does not appear, then turn it on by clicking **QUAD** on status bar or pressing the **F12** function key.

Along the bottom of the Quad is a blue bar with three words: Model, Drawing, and General. Move the cursor into one of them, such as Draw. As you do, the Quad expands to display commands related to drawing. Click an icon to start the command.



Displaying Properties with the Quad

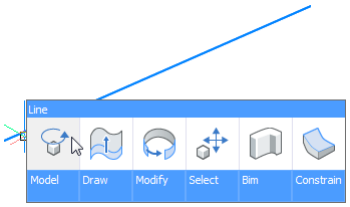
When you move the cursor over an entity, the Quad appears and reports the properties of the entity. The properties displayed can be customized by you through the **Customize** command's **Properties** tab.



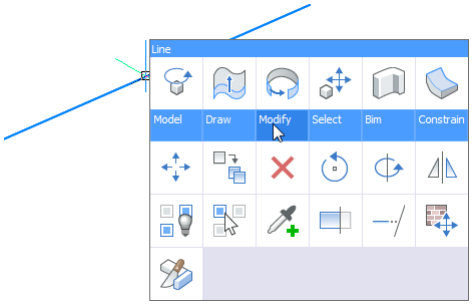
TIP If the quick properties are not displayed in the Quad, then click the **RT** (rollover tooltips) button on the status bar.

Editing with the Quad

With the Quad still hovering over the entity, move the cursor into the properties area. Notice that the Quad expands to display editing and inquiry commands.



Again, there is the blue band for groups of commands. Some groups are for common operations while others are specific to the entity. To access the additional buttons, pass the cursor over a blue band. Click a button to execute a command.



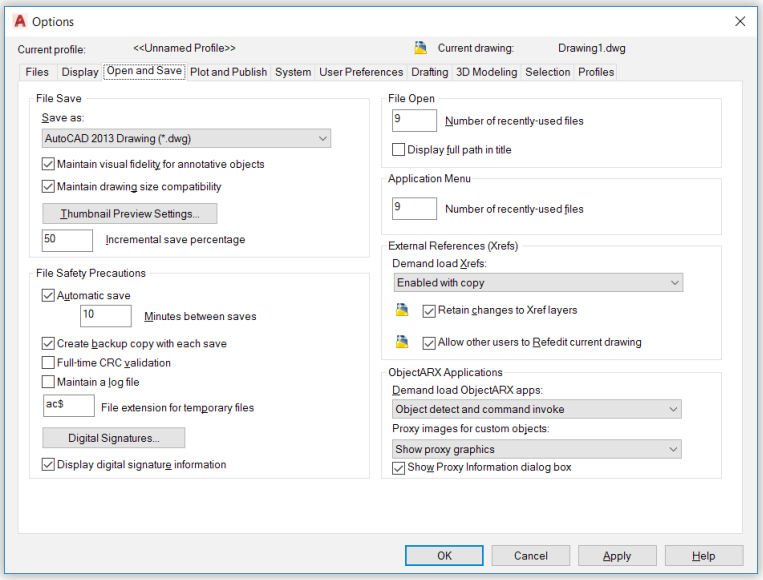
BricsCAD's Quad cursor expanding further to expose groups

BricsCAD comes with several sets of predefined Quad cursors setups, such as for 2D drafting and 3D modeling. You customize the Quad through the **Customize** command's Quad and Workspace tabs. See chapter 4.

Differences in Options, Settings

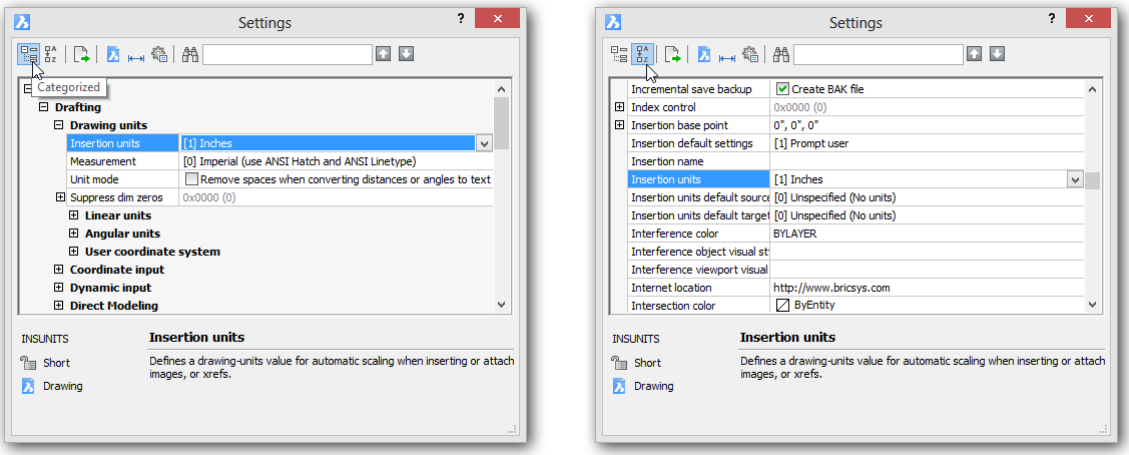
You are no doubt familiar with the Options dialog box in AutoCAD, which provides access to many system variables, through not all of them, oddly enough. In BricsCAD, the equivalent dialog box is known as **Settings**. It accesses *all* 875 variables. Appendix B provides you with the complete list in BricsCAD, along with a comparison with AutoCAD's system variables.

Providing users access to hundreds of system settings is a programming problem: how to make it easy for end users? In the case of AutoCAD, the Options dialog box is segregated into ten or eleven tabs and more than thirty auxiliary dialog boxes!



AutoCAD's Options dialog box segregates system variables into tabs, groups, dialog boxes, but does not provide access to all

In contrast, Bricsys designed a single dialog box that provides access all variables through an interactive search box. You start typing the first few characters of the name, title, or description of a variable, and BricsCAD jumps to the first instance in real time; click the arrow keys to move to additional instances of the text. Colors alert you when the text does not exist, or when you've reached the end of the instances.



Left: *BricsCAD's Settings dialog box in Category mode;* **right:** *...and in Alphabetic mode*

In BricsCAD, variables can be sorted by category or alphabetical order. Both modes are illustrated above: clicking the Categorical and Alphabetical toolbar buttons changes the sort order.

TOURING THE SETTINGS DIALOG BOX

This Settings dialog box is important to using BricsCAD effectively. Because this dialog box is designed quite differently from AutoCAD's, allow me to give you a tour of its functions. To access the Settings dialog box, use one of these methods:

- Enter the **Settings** command
- Type the **Options** alias used by AutoCAD
- From the **Settings** menu, choose **Settings**
- In the ribbon's **Home** tab, look for the **Settings** panel, and then choose **Settings**

Atop the dialog box is a toolbar from which you access BricsCAD's variables. From left to right, these buttons control the sort order, export settings, jump to major sections, and search settings by name.

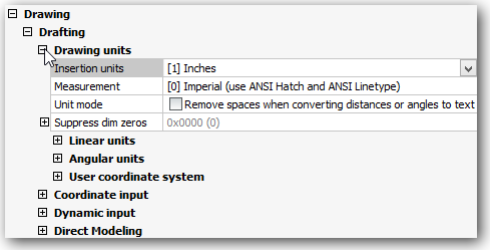
Toolbar atop the Settings dialog box

- Click either of the first two buttons to change the sorting order between Categorical and Alphabetical.
- Select the **Export** button to save setting names and values to a CSV file.
- Pick any of the next three buttons to access the **Drawing**, **Dimensioning**, or **Program Options** sections of the dialog box.
- In the **Search** field, enter text like the name or description of a variable.
- Click the arrow buttons and to jump between all instances of the text.

I use the Search field a lot, because it's the fastest way to get to a variable and change its setting.

Opening and Closing Nodes

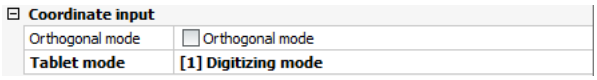
To access variables, use a method described above or else click the ⊞ boxes called "nodes" to open individual sections. (Click ⊟ boxes to close sections.) AutoCAD's CUI dialog box uses a similar system of nodes.



Opening and closing nodes to see and hide sections

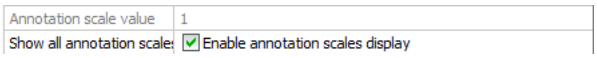
Accessing and Understanding Values

When a value is changed, it turns to **boldface** — a handy way of alerting you that change has taken place.



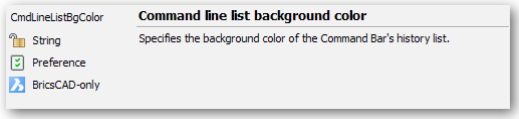
Boldfaced values have been changed since the dialog box was opened

While BricsCAD gives you access to all variables in the Settings dialog box, there are ones that you cannot change, because they are "read-only" and so are shown in gray text. Read-only variables report on the status of the system; AutoCAD also has these, but does not expose them in its Options dialog box.



Gray text indicated read-only settings

The preview area at the bottom of the Settings dialog box uses different font styles to indicate the type of variable:

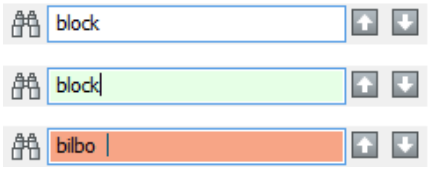


BricsCAD explaining the meaning of variables

- **UPPERCASE** text indicates system variable names, and often found in AutoCAD
- **Mixed Case** text indicates preference variable names
- icon indicates settings unique to BricsCAD, and so are not found in AutoCAD

Using Real-time Search

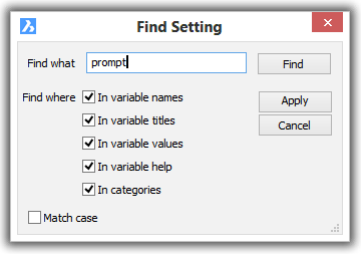
The real-time search field lets you directly access system variables by name. As you enter the first few letters, BricsCAD immediately jumps to the first name that matches them. You can then click the up and down arrows to move back and forward through matching candidates. (AutoCAD does not have a search function in its multi-tabbed Options dialog box.) The color of the search field changes to report the status of the search term you entered:



BricsCAD using colors to alert the search status

Snow white — two or more names match the search phrase
Lime green — one (or the last) name matches the search phrase
Tangerine orange — no name matches the search phrase

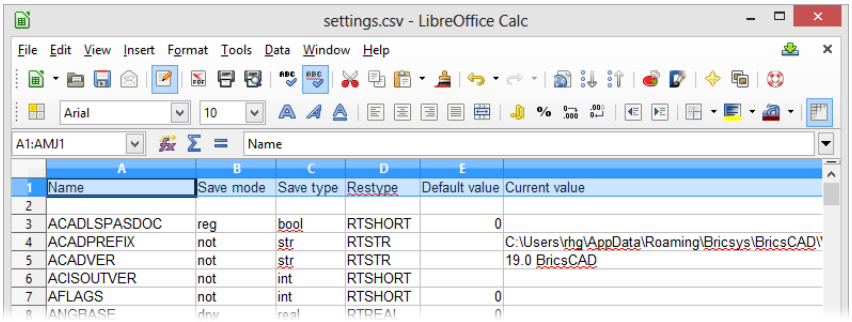
By clicking the **Find** button, you can ask BricsCAD to narrow the search, but I find it's best to leave all the Find Where options turned on. ("In variable values" is new to V17.)



Dialog box for narrowing the search field

Exporting Settings

To export the settings and their values, click the **Export** button. This action saves them to a text file formatted as CSV (comma-separated value). Such as file can be imported into LibreOffice Calc or another spreadsheet program. (AutoCAD does not provide this feature.)



	A	B	C	D	E	
1	Name	Save mode	Save type	Restype	Default value	Current value
2						
3	ACADLSPASDOC	reg	bool	RTSHORT	0	
4	ACADPREFIX	not	str	RTSTR		C:\Users\rhg\AppData\Roaming\Bricsys\BricsCAD\
5	ACADVER	not	str	RTSTR		19.0 BricsCAD
6	ACISOUTVER	not	int	RTSHORT		
7	AFLAGS	not	int	RTSHORT	0	

BricsCAD settings exported to a spreadsheet

TIP BricsCAD has the same **SetVar** command as AutoCAD for accessing variables. As in AutoCAD, you can also enter names of system and preference variables directly at the ':' prompt.

Variations in Palettes

Both CAD system offer palettes, such as Properties. BricsCAD uses the word "panel" in place of palette. Items shown in blue are new to V17. BricsCAD's Drawing Explorer is not a panel or palette. Here is the list of panel-palettes provided:

AutoCAD Palette	BricsCAD Panel	Notes
Advanced Render Settings	(Drawing Explorer)	BricsCAD handles render settings in the Drawing Explorer
...	BIM Composition panel	BIM models are not supported by AutoCAD
Command	Command bar	
dbConnect	...	Database linkages not supported by BricsCAD
DesignCenter	Content Browser panel	BricsCAD also handles this through Drawing Explorer
External References	(Drawing Explorer)	BricsCAD handles references in Drawing Explorer
Layer	Layers panel	BricsCAD also uses a dialog box for layers
Lights	(Drawing Explorer)	BricsCAD handles lights in Drawing Explorer
Markup Set Manager	...	Markups not supported by BricsCAD
Materials Browser	Render Materials panel	
Materials Editor	(Drawing Explorer)	BricsCAD edits materials in Drawing Explorer
Parametrics	Mechanical Browser panel	
Properties	Properties panel	
QuickCalc	...	
Ribbon	Ribbon panel	
Sheet Set Manager	Sheet Sets panel	
...	Structure panel	AutoCAD does not have a drawing structure browser
Tool Palettes	Tool Palettes panel	
Visual Styles	(Drawing Explorer)	BricsCAD handles visual styles in Drawing Explorer

In the following sections, we look at some panels that are the similar in both CAD systems — Properties, Layers, and Sheet Sets, and Mechanical Browser panels— and a couple that are unique to BricsCAD: Content Browser and Structure.

PROPERTIES PANELS

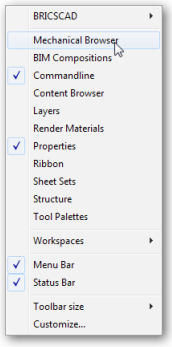
The two CAD packages share a similar-looking Properties palette, except that BricsCAD calls its the Property "panel." It operates just like the Properties palette in AutoCAD, but with this important difference: BricsCAD employs the Properties panel for all editing functions and changes to properties in those areas where AutoCAD tends to display command-specific dialog boxes or bring up contextual tabs on the ribbon. Naturally, both also employ grips editing as well as specific editing commands.

For example, when you click on a hatch pattern in BricsCAD, the Properties panel displays all the options you expect to find in AutoCAD with its Hatch Edit dialog box and contextual ribbon.

To turn on the Properties panel in BricsCAD, enter the **Properties** command. The panel appears automatically when you double-click entities in drawings.

ACCESSING AND MOVING BRICSCAD PANELS

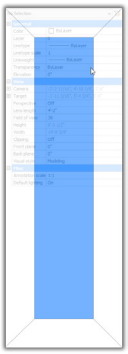
You can open and close BricsCAD panels with commands, but the easiest way to access them is by right-clicking any other user interface element, such as the ribbon or a toolbar. Choose a name from the shortcut menu:



OVERLAPPING PANELS

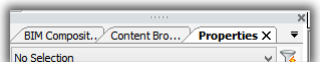
(NEW TO V17) BricsCAD has nine panels, and so when many of them are open, they take up a lot of screen real estate. One solution is to park them on a second monitor. Another solution is to overlap them, as follows:

1. Drag a panel over top another one.
2. Notice the blue trapezoids that appear.



Each refers to a location:

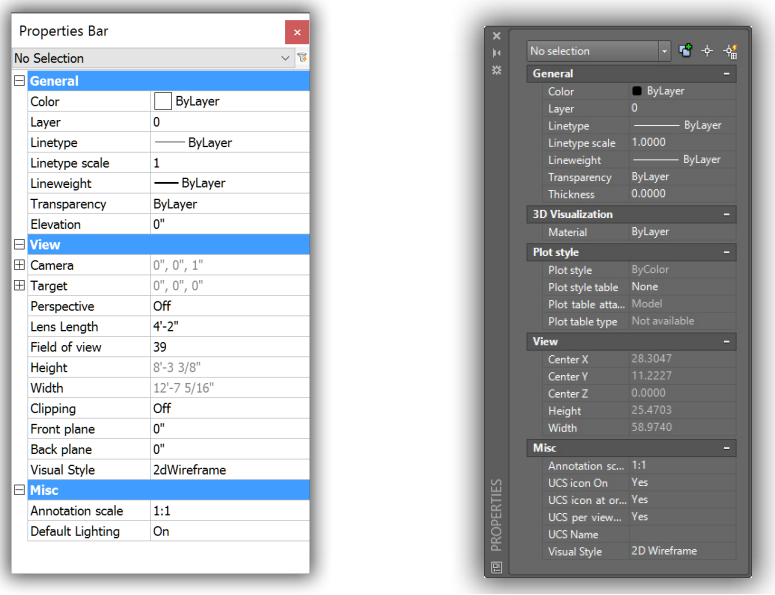
- Top and bottom trapezoids — panel is parked to the top (or bottom) of existing ones
- Side trapezoids— panel is parked at the side of the existing one(s)
- Center trapezoid — panel is turned into a tab, as illustrated below



3. Move the panel into one of the trapezoids.

AutoCAD stacks multiple palettes to the side of the screen.

As in AutoCAD, you can in BricsCAD assign double-click actions to entities, which then display the Properties panel with the parameters appropriate to the entity. (See chapter 4 more on this.)

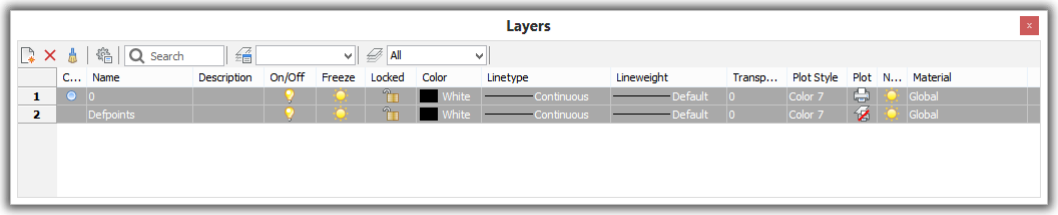


Left: *Properties panel in BricsCAD;* ***right****: Properties palette in AutoCAD*

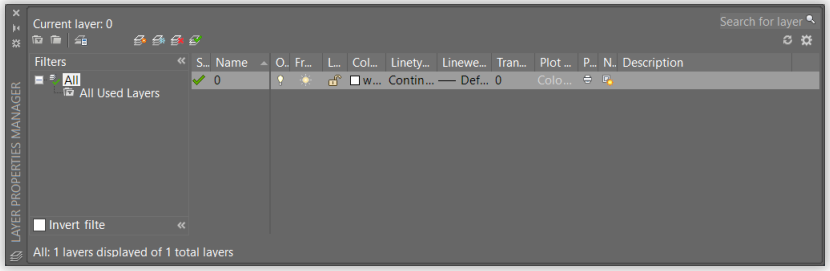
(NEW IN V17) Geometric properties for surfaces — such as Watertight, Loops, Holes, Lumps, and Faces — are properties added to the Properties panel. Read-only values can be copied to the Clipboard.

LAYERS PANELS

(NEW TO V17) AutoCAD and BricsCAD both report layer names, status, and properties in a dialog box, a panel, and in droplists on toolbars and the ribbon. Use the **LayerPanelOpen** and **LayerPanelClose** commands to open and close the panel in BricsCAD.

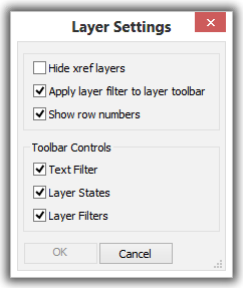


Above: *Layers panel in BricsCAD;* ***below:*** *Layer Properties Manager palette in AutoCAD*



The toolbar of the Layers panel in BricsCAD performs the following functions:

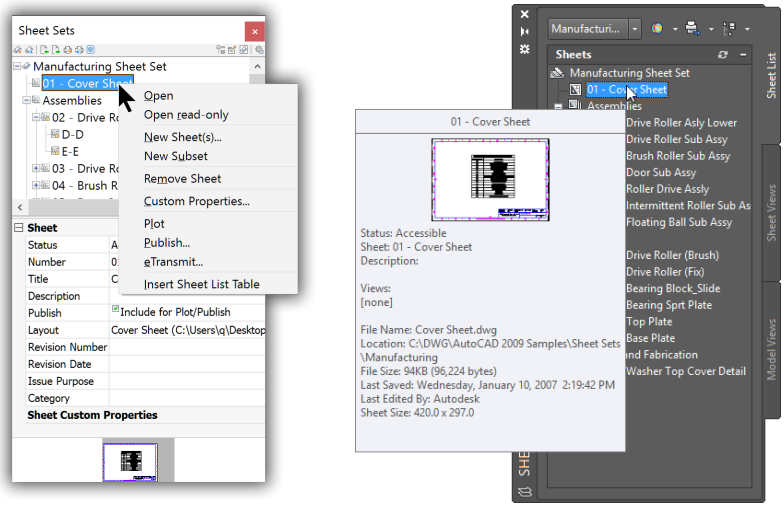
- New layer
- Remove layer
- Purge unused layers
- Layer settings



- Search for a layer name
- Open the Layer States node in the Drawing Explorer
- Select layer state droplist
- Open the Layer node in the Drawing Explorer
- Select layer filter droplist

SHEET SETS

BricsCAD supports sheet sets, although the number of functions is fewer than in AutoCAD. The figures below show an AutoCAD sample sheet set opened in BricsCAD.



Left: *BricsCAD's sheetset manager;* **right:** *AutoCAD's sheetset manager*

BricsCAD's Sheet Set user interface looks like AutoCAD's, a single palette. Both CAD programs use right-click menus and a toolbar to create, edit, and publish sheet sets.

To create and control sheet sets in BricsCAD, enter the **SheetSet** command, click the **Sheets** tab, and then choose from among the buttons on the toolbar:

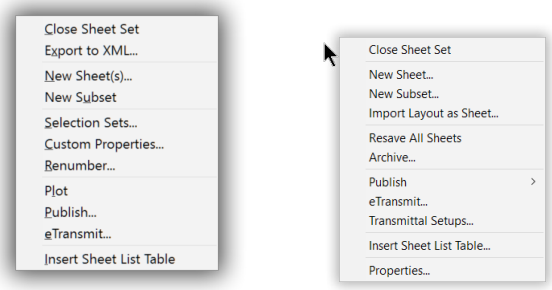
Left: *Sheet set toolbar in BricsCAD;* **right:** *Sheet set toolbar in AutoCAD*

From left to right in BricsCAD, the buttons perform the following functions:

- Create a new sheetset using a wizard (**NewSheetSet** command)
- Open a DST file, which defines an existing sheetset (**OpenSheetSet** command)
- Import from XML
- Export to XML
- Print the selected drawing (**Plot** command)
- Publish the sheetset (**Publish** command)
- Bundle the sheetset for transmittal by email (**eTransmit** command)
- Create a sheetset selection set
- Create custom properties
- View categories
- Sheet set options (**Options** command)

Missing from BricsCAD are archives.

The shortcut menus shown below illustrate the differences in capabilities.



Left: *Sheetset shortcut menu in BricsCAD;* **right:** *Sheetset shortcut menu in AutoCAD*

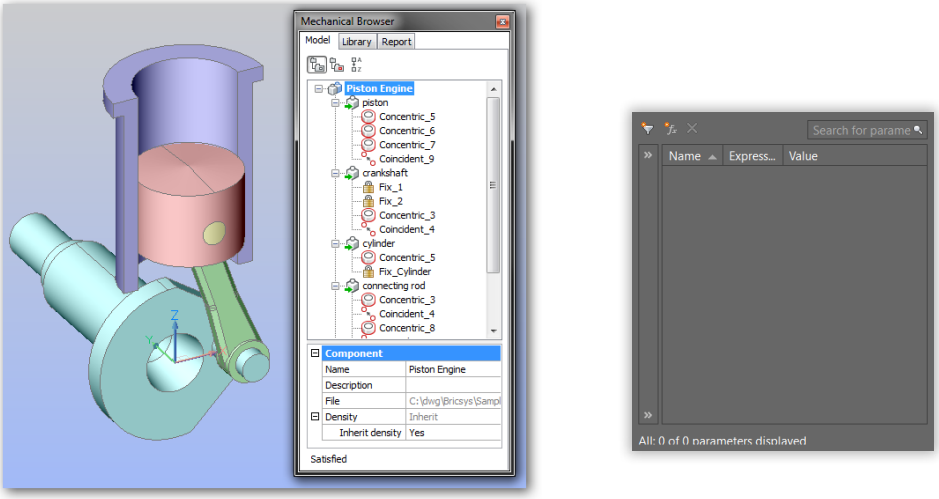
BricsCAD uses the same DST format as AutoCAD's sheet sets, and so you can reuse them from AutoCAD. In addition, BricsCAD imports and exports sheet set files in XML format, and prints sheet sets with the Publish command.

MECHANICAL BROWSER VS PARAMETRICS MANAGER

Both CAD systems provides parametrics constraints, but here BricsCAD outdoes AutoCAD. This table illustrates the differences:

Feature	BricsCAD	AutoCAD
2D geometric constraints	12	12
2D dimensional constraints	8	6
3D geometric constraints	7	0
3D dimensional constraints	3	0
Formulas in constraints	Yes	Yes
Assemblies from parts	Yes	No

The Mechanical Browser in BricsCAD shows the sophistication of its 3D parametric modeling capabilities. (Three-D constraints are not available in AutoCAD.)



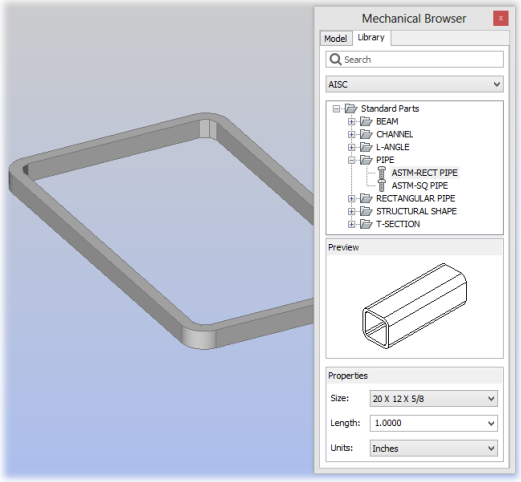
Left: *BricsCAD's Mechanical Browser handles constraints, parameters, and assembly parts;* ***Right:*** *AutoCAD's Parametrics Manager with constraint formulas only*

Constraints added to models in BrisCAD are not recognized in AutoCAD. BricsCAD, however, reads constraints from AutoCAD drawings due to the ODA Teigha library. AutoCAD uses the constraint engine from Siemens PLM Software; BricsCAD uses the constraint engine it developed itself.

(new in V17) 3D constraints now take lines, circles, arcs, xlines, and rays as arguments. If a constraint of the same type already exists, the new constraint is created with the "Disabled" flag.

Parts Library (BricsCAD Only)

To assist with 3D modeling, BricsCAD includes a library of parametric parts. Choose a part from the tree in the Mechanical Browser, adjust the size in the Properties pane, and then drag the part into the drawing, where additional prompts appear in the command bar to insert and rotate the part.

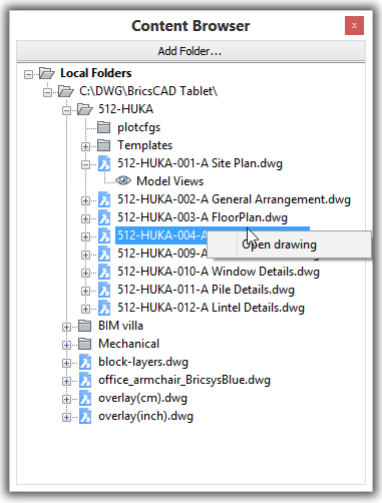


Parts library in BricsCAD

TIP Tool palettes operate similarly in both CAD programs, except that BricsCAD stores them in BTP files (short for "BricsCAD tool palettes"), whereas AutoCAD stores tool palette definitions in ATP files (short for "AutoCAD tool palettes") . Both file types use XML as the format. See chapter 4 on customizing Tool Palettes in BricsCAD.

CONTENT BROWSER PANEL (BRICSCAD ONLY)

(NEW TO V17) The Content Browser panel shows a tree view of drawings found in folders (as specified by you). CLick the Add Folder button to select folders on your computer, on networked computers, and cloud storage services, such as Dropbox.



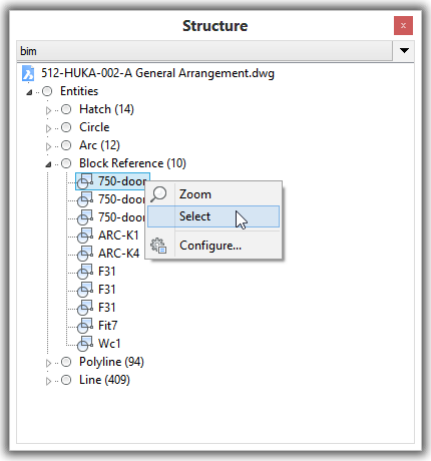
Content Browser showing drawings in user-specified folders

Double-click a file name to open the drawing in a new window. Single-click a file name to shows model space views, which can be dragged into the current drawing. Dragging model views from the Content Browser activates the new Placeview command automatically. Bricsys plans to add more drawing content in future releases, such as blocks and dimension styles.

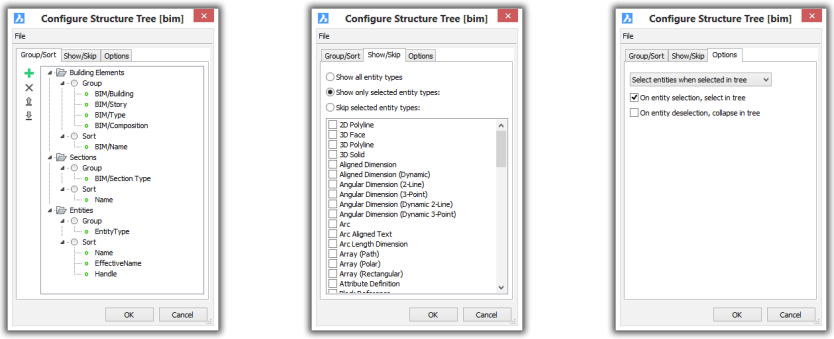
Use the **ContentBrowserOpen** and **ContentBrowserClose** commands to open and close the .

STRUCTURE PANEL (BRICSCAD ONLY)

(NEW TO V17) he Structure panel displays a structured tree view of the drawing's content. When entities are selected in the structure tree, they are highlighted, zoomed, and selected in the drawing — and vice versa. The panel operates in model space only.



The format of the panel is customized through the Configure dialog box, and then saved and loaded through *.cst* configuration files. Bricsys provides three .cst files in the *C:\Users\userid\AppData\Roaming\Bricsys\BricsCAD\V17x64\en_US\Support* folder: BIM, Mechanical, and Default.



Configuring the Content Browser panel

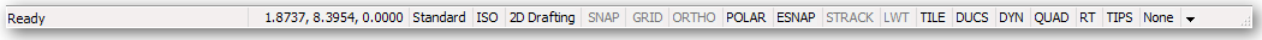
TIP Use the Content Browser to access drawings *outside* of BricsCAD; use the Structure panel to access content in drawings *inside* of BricsCAD.

Status Bar & Other UI Differences

Overviews of the differences in additional user interface elements: the status bar, working sets (BricsCAD only), selection sets, DesignCenter vs Drawing Explorer, and Autodesk 360 vs Chapoo.

DIFFERENCES IN STATUS BARS

The status bar in BricsCAD reports the status of the drawing, just like in AutoCAD. The two have a few differences in the functions they provide. BricsCAD continues to use text for the buttons, while AutoCAD shows users icons by default, which can be confusing for new users.



Above: *Default status bar in AutoCAD;* ***below:*** *Default status bar in BricsCAD*

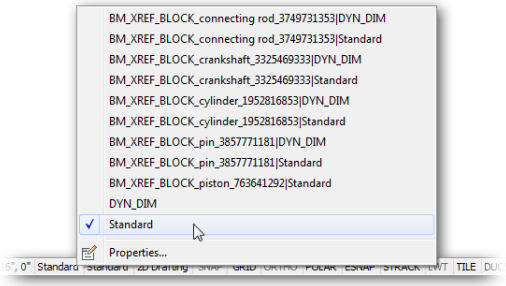
This is the list of similarities and differences of the contents of the two status bars:

Status Bar Function	AutoCAD	BricsCAD	Notes
Diesel prompts	Yes	Yes	Through the ModeMacro command
Command prompts	...	Yes	When command bar is turned off
Cursor coordinates	Yes	Yes	
Current layer name	...	Yes	
Current color	...	Yes	
Current linetype	...	Yes	
Current text style	...	Yes	
Current dimension style	...	Yes	
Workspaces	Yes	Yes	
Snap toggle	Yes	Yes	
Grid toggle	Yes	Yes	
Ortho toggle	Yes	Yes	
Polar toggle	Yes	Yes	
Entity Snap toggle	OSnap	ESnap	
Object Tracking	OTrack	STrack	
Lineweight toggle	LWT	LWT	
Model / Tile	Yes	Yes	
Annotation Scale	Yes	Yes	
Annotation Visibility	Yes	...	
AutoScale	Yes	Yes	
Tablet	...	Yes	
Dynamic UCS	DUCS	DUCS	
DYN	Yes	Yes	Dynamic input toggle
QUAD	...	Yes	Quad cursor toggle
RT	Yes	Yes	Rollover Tooltips
Tips	...	Yes	
GIS Coordinate System	(Yes)	Yes	AutoCAD displays geo coordinates in Coordinates field

Additional status items with AutoCAD:

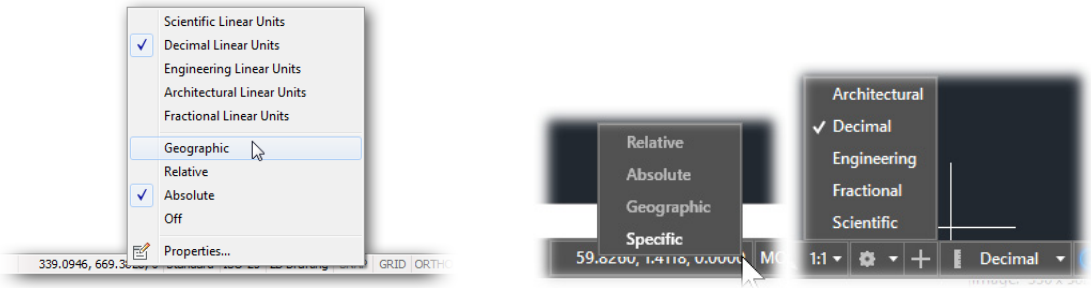
Infer Constraints	Yes	...	BricsCAD has design intent
Isometric Drafting	Yes	...	BricsCAD has isometric mode
Transparency	Yes	...	BricsCAD sets transparency through variables
Selection Cycling	Yes	...	BricsCAD cycles through selections
Selection Filtering	Yes	...	BricsCAD has a selection cycling toolbar
Gizmo	Yes	...	BricsCAD does not have the 3D editing gizmo
Units	Yes	(Yes)	BricsCAD settings is in Coordinates shortcut menu
Quick Properties	Yes	...	BricsCAD does not have Quick Properties
Graphics Performance	Yes	...	BricsCAD uses variables to set graphics performance
Clean Screen	Yes	...	BricsCAD does not have a clean screen function

As in AutoCAD, you right-click a toggle on the BricsCAD status bar to access options. BricsCAD, however, goes one step further: to change a text or dimension style, just right-click the current name, and then choose a different one from the shortcut menu. (AutoCAD does not offer this function.)



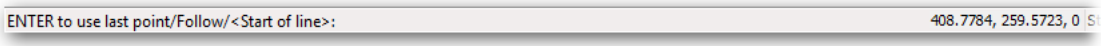
Accessing dimensions styles from the status bar in BricsCAD

All coordinate options are accessed from a single status bar button, while AutoCAD requires two buttons for the same job.



***Left:** Accessing units formats from the status bar in BricsCAD; **right:** AutoCAD requiring two status bar buttons to do the same*

Right-clicking the at the right end of the status bar produces a menu in BricsCAD and AutoCAD. It controls the items seen on the status bar. The BricsCAD status bar does double duty: when the command bar is turned off, the program's prompts appear on the status bar. (AutoCAD does not provide this function.)



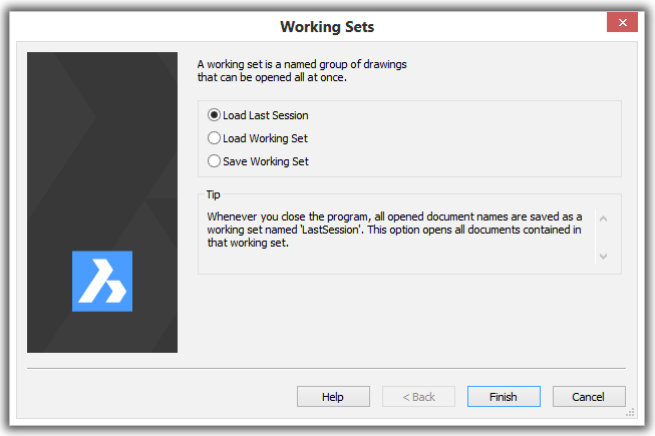
Status bar in BricsCAD displaying command prompts

WORKING SETS (BRICSCAD ONLY)

Working sets group drawings by name. With this name, you load two or more drawings simultaneously into BricsCAD. The **Workset** command is useful, because Bricsys implemented threaded file opening, which uses the computer's multi-core CPU to perform more than one task at a time; the command is also necessary for BricsCAD's assembly function, which loads multiple drawings of parts. (AutoCAD cannot load multiple drawings at the same time, although a workaround is to use sheet sets.)

When you close BricsCAD, it saves the names of all open drawing files automatically as a working set under the generic name of "LastSession." This means you can you easily open all previous drawings the next time you start BricsCAD.

After BricsCAD opens, you access worksets through the **Workset** command.



Dialog box for loading and saving working sets

TIPS WIDGET (BRICSCAD ONLY)

"Tips" are like interactive bats. (AutoCAD has nothing like this.) They report command options that might otherwise be unknown to users. For example, the following Tips widget appears during the Polysolid command.

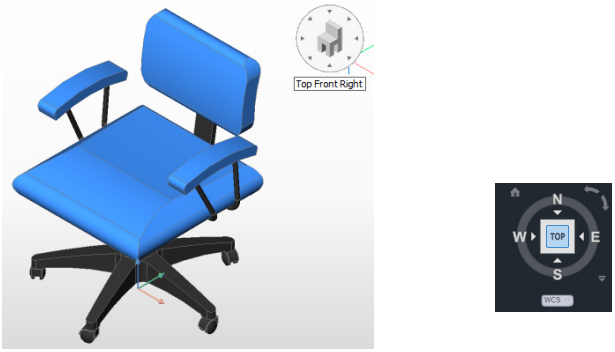
The Tips widget for the Polysolid command

Tips widgets show several icons. They indicate that by holding down the **Ctrl** key during the command, the user can change the justification between left, centered, and right.

The display is toggled through the **TIPS** button on the status bar. Pause the cursor over the Tip to get a brief description of the purpose. Click the **x** to dismiss the Tip.

DIFFERENCES IN VIEW CUBES

AutoCAD has the navigation cube for quickly changing 3D viewpoints; in BricsCAD, it is known as the LookFrom widget. Passing the cursor over the small triangles displays the preview of a chair; clicking the triangle changes the 3D viewpoint. Hold down the **Ctrl** key to see the bottom views.

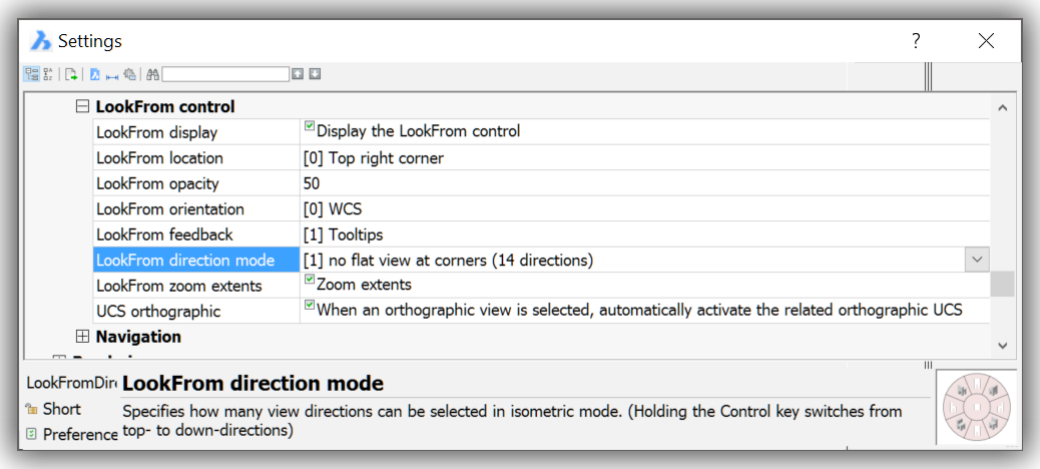


Left: *LookFrom control in BricsCAD;* ***right:*** *ViewCube control in AutoCAD*

There are two ways to change the way the LookFrom control operates. One is to enter the **LookFrom** command, from which you can turn it off (and on) or access its settings:

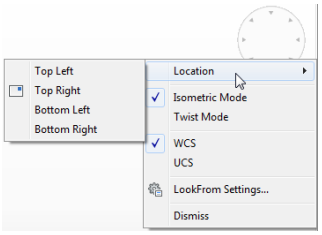
```
: lookfrom
LookFrom [ON/OFF/Settings] <ON>:
```

Turn it off for 2D drafting. The **Settings** option opens the Settings dialog box at the LookFrom section. Here you adjust the properties of the widget, such as its translucency and the number of isometric viewpoints it displays (Direction Mode).



LookFrom properties in the Settings dialog box

The other method is to right-click the control, and then choose an option.



Context menu for the LookFrom control

The difference between Isometric Mode and Twist Mode rotating the 3D viewpoint:

- Isometric mode is like the Viewpoint or View commands
- Twist mode is like the RtRotF (3DOrbit) command

The green dot indicates the cursor position, kind of like a laser pointer:

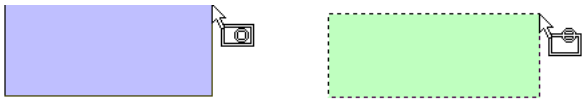
Left: *Isometric mode*
Right: *Twist mode*

TIP When in Twist mode, click the center of the LookFrom control to return the view to its home view.

DIFFERENCES IN SELECTION SETS

You can assemble complex selection sets in BricsCAD through entity location (pick, Window, Crossing, and so on) and/or properties (color, linetype, and so on), as in AutoCAD. Many actions are the same between the two CAD programs, such as pressing **Ctrl+A** to select all objects in drawings. Like AutoCAD, BricsCAD makes sub-entity selection of 3D objects: faces, edges, and vertices.

Like AutoCAD, BricsCAD uses colors to report to the user whether the current selection set is a crossing, window, or other. Unlike AutoCAD, however, BricsCAD also displays icons, as shown below. (The closest AutoCAD has to these icons are *cursor badges*, which show which command is effect.)

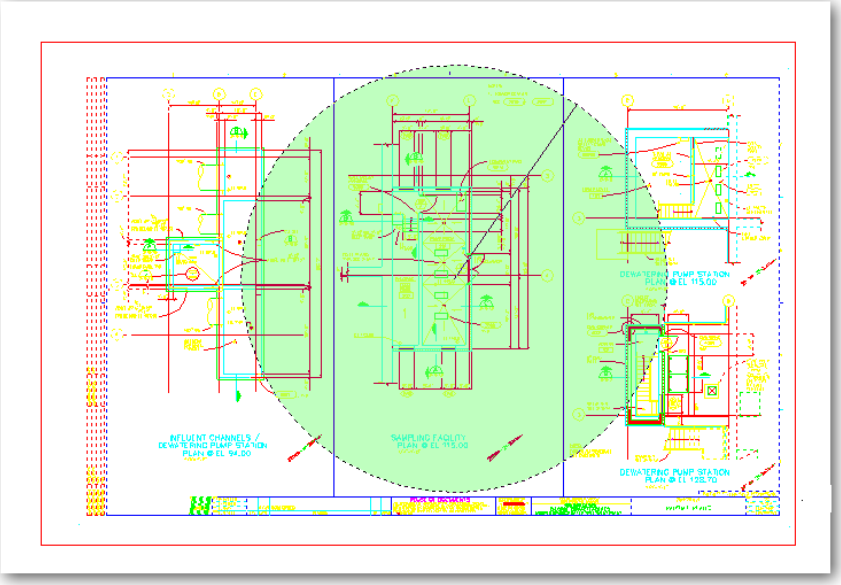


BricsCAD uses colors and icons to report the style of a windowed selection:
Left: *Making a windowed selection.*
Right: *Making a crossing selection.*

BricsCAD's Select command displays the names of options when you enter '?'. AutoCAD's Select command does not, except by a workaround (enter the name of a non-valid option). Here is the BricsCAD version of the command:

```
: select
Select entities to include in set: ?
Select entities:  ALL/Add/+/Remove/-/Previous/Last/Window/Crossing/Outside/WPolygon/CPoly-
gon/OPolygon/WCircle/CCircle/OCircle/Box/POint/Fence/AUto/Multiple/Single/PROperties/Dialog/
Undo/Group:
```

AutoCAD 2015 added the lasso selection mode, not found in BricsCAD. On the other hand, BricsCAD has these selection modes not found in AutoCAD:



BricsCAD selecting all objects inside a circular selection window

- **Outside window** (O) — selects all entities fully outside of a rectangular window
- **Outside polygon** (OP) — selects all entities fully outside of an irregular polygon

- **Window circle** (WC) — selects all entities fully within a circle
- **Crossing circle** (CC) — selects all entities within and crossing a circle; see figure below
- **Outside circle** (OC)— selects all entities fully outside of a circle

The Dialog option displays the Settings dialog box for making changes to selection settings.

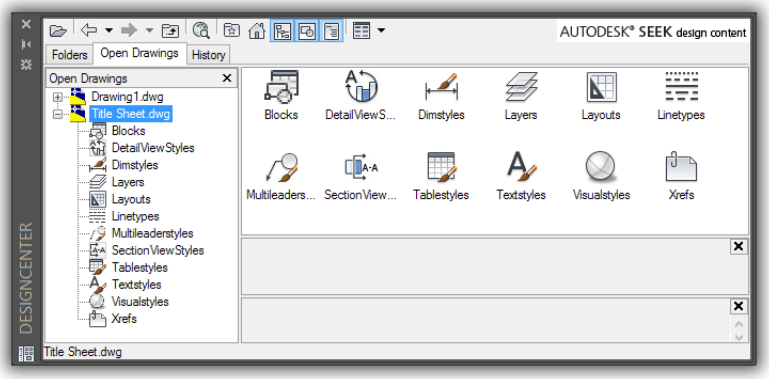
VARIATIONS IN DESIGNCENTER & DRAWING EXPLORER

BricsCAD's Drawing Explorer is best compared with AutoCAD's DesignCenter, but Explorer reports more information and provides greater control over drawing elements. Drawing Explorer centralizes in BricsCAD what in AutoCAD amounts to as many separate dialog boxes; facilities such as layer management, UCS control, and control of external references are in one location. (Autodesk appears to be copying BricsCAD by amalgamating similar commands, such as Attach.)

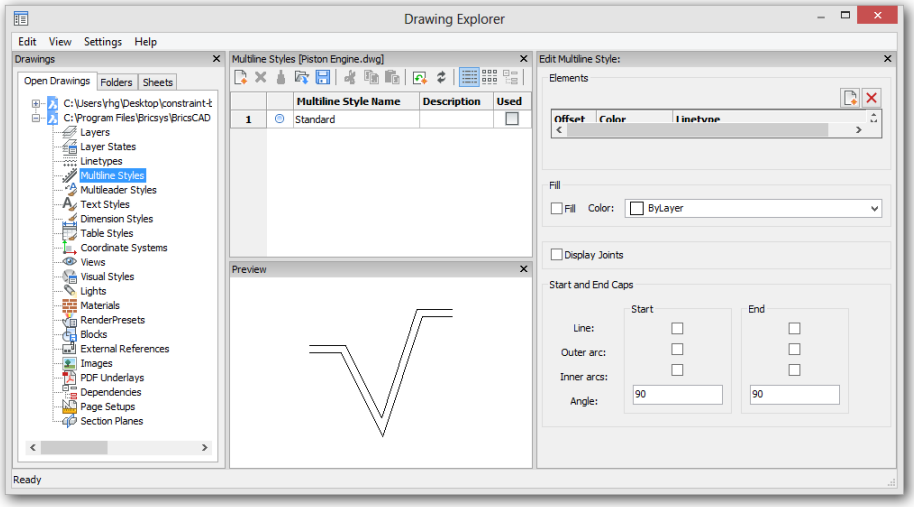
Drawing Explorer handles all named entities, and these are listed in the table below.

AutoCAD's DesignCenter Node	BricsCAD's Drawing Explorer Node	Alternate Commands
Blocks	Blocks	
...	Chapoo (in Folders tab)	SaveToCloud command (Autodesk 360)
...	Coordinate Systems	UcsMan command in AutoCAD
...	Dependencies	eTransmit command in AutoCAD
DetailViewStyles	...	ViewDetailStyle command in BricsCAD
Dimstyles	Dimension Styles	
Xrefs	External References	
...	Images	ExternalReferences command in AutoCAD
Layers	Layers	
...	Layer States	LayerStates command in AutoCAD
...	Lights	LightList command in AutoCAD
Linetypes	Linetypes	
...	Materials	MatBrowserOpen command in AutoCAD
Layouts	Page Setups	
...	Multiline Styles	MlStyle command in AutoCAD
Multileaderstyles	Multileader Styles	
...	PDF Underlays	PDFAttach command in AutoCAD
...	Render Presets	RenderPresets command in AutoCAD
...	Section Planes	SectionPlaneSettings command in AutoCAD
SectionViewStyles	...	ViewSectionStyle command in BricsCAD
Tablestyles	Table Styles	
Textstyles	Text Styles	
...	Views	View command in AutoCAD
VisualStyles	Visual Styles	

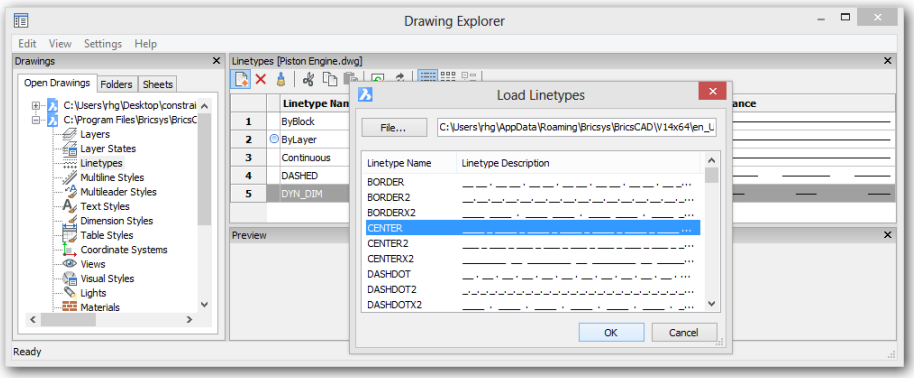
To access BricsCAD's Drawing Explorer, enter the **Explorer** command. BricsCAD displays Drawing Explorer automatically when you enter related commands, such as Layer or Xref.



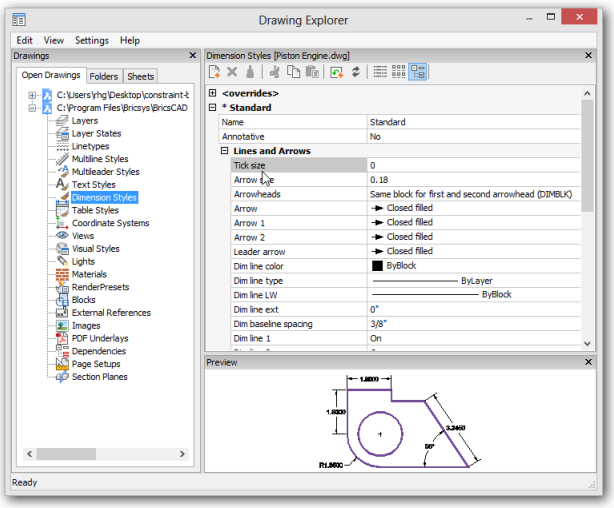
Above: *AutoCAD's Design Center.*
Below: *BricsCAD's Drawing Explorer.*



BricsCAD includes settings for modifying these named entities, something lacking in AutoCAD's DesignCenter. For example, the Linetypes node lets you load additional linetypes:



... and the Dimension Styles node lets you modify the styles:



BricsCAD creating, modifying, and applying dimensions styles

Unified Interface

Drawing Explorer is more than a DesignCenter because it centrally gathers commands for inserting and controlling named entities. This is the same philosophy that drives Bricsys to make the Settings dialog box access all system variables, instead of just some of them.

By my count, the unified interface of BricsCAD's Drawing Explorer replaces the equivalent of 23 AutoCAD commands and related dialog boxes and palettes.

3D MODELING

See Chapter 6 for the differences between AutoCAD and BricsCAD in the area of 3D modeling.

CHAPOO VS 360

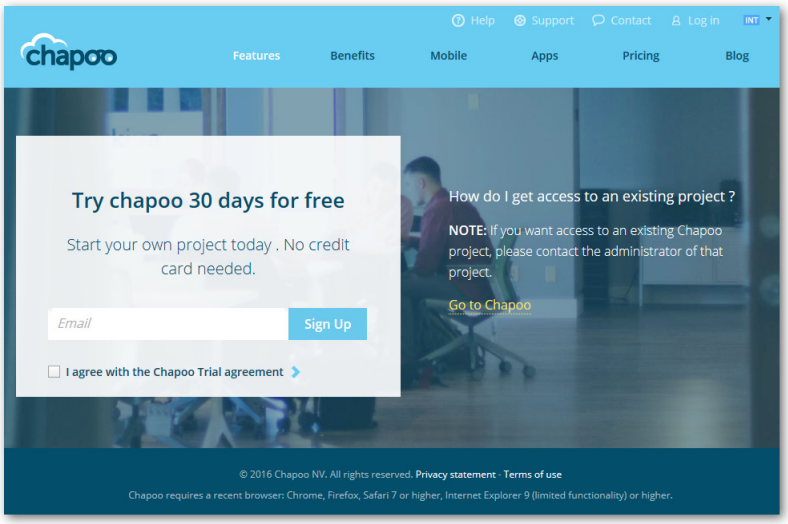
Chapoo is the online collaboration and cloud storage from Bricsys. The equivalent in AutoCAD is Autodesk 360. Commands inside BricsCAD let you open and save files from and to the Chapoo cloud.

Within Chapoo, you create collaboration areas,which are helpful for project management. Chapoo provides the following services:

- Project collaboration through project-specific email, forums, and data repositories
  - Version control through check-in/checkout
  - Calendar and address book for each project
  - Document management with sharing, viewing, and markups
  - Document viewing of 70+ file formats

- Project administration for assigning rights, folders, and so on
  - Access control assigned to managers, contractors, customers, supplies, and so on
  - Live data created from forms and data (optional add-on)
  - Graphical workflows created through a drag-and-drop editor

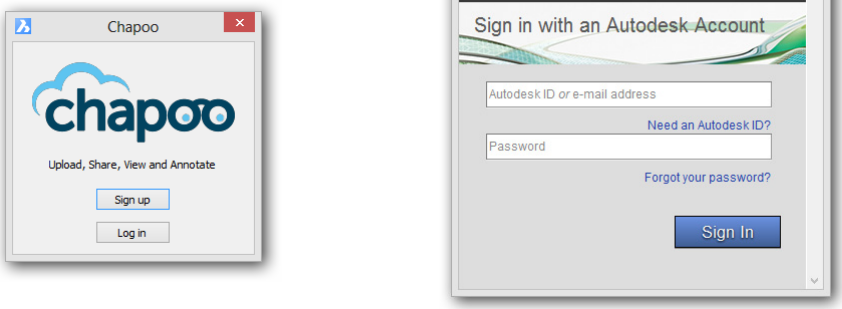
To sign up for the 30-day free version of Chapoo, go to http://chapoo.com/en_INTL.



Accessing Chapoo for the first time

Using Chapoo

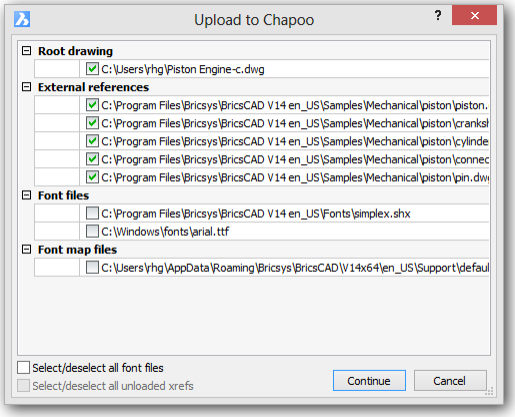
To log into Chapoo from BricsCAD, enter the **ChapooOpen** command.



***Left:** Logging into Chapoo from BricsCAD; **right:** Logging into 360 from AutoCAD*

Commands in BricsCAD let you upload and download files:

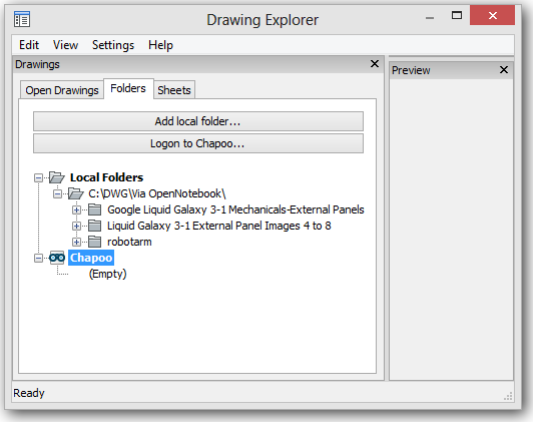
ChapooOpen opens files stored online
ChapooDownload downloads files from online to your computer
ChapooUpload uploads the current file to your online account, along with all dependent files, such as xrefs and image files, and optionally uploads fonts



Checking dependent files for drawing being uploaded to Chapoo

ChapooProject switches to the Web browser, and then opens your Chapoo account online
ChapooWeb also switches to the Web browser, and then opens the Chapoo home page
ChapooLogoff logs out of your Chapoo account

With the connection made between your computer and Chapoo, your files are made available through the Folders tab of Drawing Explorer. (This place is an alternative location for logging into Chapoo.)



Accessing your folders on Chapoo

— — —

THIS CHAPTER HIGHLIGHTED the differences in the user interface of BricsCAD and AutoCAD. Many of them are identical or similar, but some elements in BricsCAD are unique The next chapter examines how both programs display and edit entities in drawing files.

CHAPTER THREE

Compatibility of Drawing Elements

BRICSCAD READS AND WRITES AUTOCAD DRAWINGS VERY WELL, BUT IN A FEW CASES NOT perfectly.

For mixed-CAD offices or BricsCAD design firms working in a DWG world, it is crucial that the two CAD systems exchange drawings accurately. Use this chapter to assist you in pinpointing problem areas, should any occur.

This chapter details how well BricsCAD does at reading entities, properties, and styles created by AutoCAD. The two CAD programs handle a large range of DWG and DXF versions, but BricsCAD does better than AutoCAD with older ones. Use the **Open** and **SaveAs** commands to access DWG and DXF files in the following versions:

Format	BricsCAD	AutoCAD
Oldest DWG format	Release 12 (from 1993)	Release 14 (from 1997)
Oldest DXF format	Release 9 (1987)	Release 12 (1993)
Newest DWG/DXF format	Releases 2013 - 2017	Releases 2013 - 2017

In summary, BricsCAD V17 reads and writes all the same DWG and DXF files as AutoCAD does, but goes further back in time. This is useful when working with archived drawings from projects initiated in the late 1980s and early 1990s.

TIP While Autodesk for some time has changed the DWG file format every third year, it did not do so with AutoCAD 2017. Autodesk can, however, fit in new entities, properties, and styles with any release. The current DWG version is R20.0.

Entity Types

This chapter graphically illustrates the accuracy of BricsCAD's ability to read, display, and edit the many types of entities found in DWG 2013-7 files. See the boxed text on the facing page for the complete list.

There is more DWG to just displaying AutoCAD drawings accurately. BricsCAD must display entities that come in a variety of modes, such as different styles of points and kinds of 3D surfaces. It must be able to draw and edit them in a variety of ways — such as mtext and tables. And it must handle properties and tables correctly, as described next.

Working with drawing files defined by Autodesk is a complex task that BricsCAD handles admirably.

Properties

The look of entities is controlled by *properties,* and so this chapter reports on the accuracy of BricsCAD's ability to read, display, and write the following properties found in DWG 2013-7 files:

- **Properties:** annotative scaling, colors (BYLAYER, BYBLOCK, ACI colors, and True Colors), elevations, hyperlinks, linetypes and linetype scales, lineweights, materials, plot styles, thicknesses, and transparencies
- **Layers:** status, name, on/off, freeze/thaw, lock/unlock, color, linetype, lineweight, transparency, plot style, plot, new viewport (VP), freeze new VP, VP freeze current VP, VP color, VP linetype, VP lineweight, VP transparency, and VP plot style description

Styles or Tables

Styles specify properties to specific entities by a single name. In the DWG/DXF definition, styles are called "tables," even though they have nothing to do with table entities. This chapter describes how well BricsCAD handles the following styles:

- Detail view styles and section view styles
- Dimension styles
- Multiline leader styles
- Mtext and text styles
- Multiline styles
- Plot styles
- Section styles
- Table styles
- Visual styles

CHECKLIST OF DWG 2017 ENTITIES

The following checklist shows the names all entities supported by DWG 2017. Those with gray filled-in boxes are specific to dynamic blocks.

- 2D Polyline
- 3 Point Angular Dimension
- 3D Face
- 3D Polyline
- 3D Solid
  - Box
  - Cone
  - Cylinder
  - Pyramid
  - Sphere
  - Torus
  - Wedge
  - Extrusion
  - Sweep
  - Revolve
  - Loft
- ACADPROXY_ENTITY
- Aligned Constraint Parameter
- Aligned Dimension
- Alignment Grip
- Alignment Parameter
- Angular Constraint Parameter
- Angular Dimension
- Arc
- Arc Length Dimension
- Array (Path)
- Array (Polar)
- Array (Rectangular)
- Array Action
- Attribute
- Attribute Definition
- Base Point Parameter
- Block Properties Table
- Block Reference
- Block Table Grip
- Body
- Circle
- Detail Boundary
- DGN Underlay
- Diameter Constraint Parameter
- Diametric Dimension
- Drawing View
- DWF Underlay
- Ellipse
- External Reference
- Flip Action
- Flip Grip
- Flip Parameter
- Geomap Image
- Hatch
- Helix
- Horizontal Constraint Parameter
- Jogged Dimension
- Leader
- Light
- Line
- Linear Grip
- Linear Parameter
- LINEARCONSTRAINTPARAMETERENTITY
- Lookup Action
- Lookup Grip
- Lookup Parameter
- Mesh
- MInsert Block
- MLine
- Move Action
- MText
- Multileader
- OLE
- Ordinate Dimension
- PDF Underlay
- Point
- Point Cloud
- Point Parameter
- Polar Grip
- Polar Parameter
- Polar Stretch Action
- Polyface Mesh
- Polygon Mesh
- Polyline
- Position Marker
- Radial Dimension
- Radius Constraint Parameter
- Ray
- Region
- Rotate Action
- Rotated Dimension
- Rotation Grip
- Rotation Parameter
- Scale Action
- Section Line
- Section Object
- Shape
- Solid
- Spline
- Standard Grip
- Stretch Action
- Surface
  - Extrusion
  - Loft
  - NURBS
  - Planar
  - Revolve
  - Sweep
- Table
- Text
- Tolerance
- Trace
- Vertical Constraint Parameter
- Viewport
- Visibility Grip
- Visibility Parameter
- XLine
- XY Parameter

DWG 2013 - 2017 Compatibility

With each release of BricsCAD, Bricsys adds supports more entities and properties created by AutoCAD. While BricsCAD displays all entities in drawings created by AutoCAD, it does not, however, necessarily create or edit all of them. This chapter provides details on the entities and properties that work fully and those that don't.

HOW WE TEST ENTITY COMPATIBILITY

To test BricsCAD's compatibility with AutoCAD's entities, we employed the following procedure:

1. Draw entities in AutoCAD, and then saved them to a DWG file.
2. Open the DWG file in BricsCAD V17.
3. Examine each entity for the following characteristics:
   - **Translation** — did the entity appear in BricsCAD?
   - **Visual accuracy** — does the entity look the same in BricsCAD as in AutoCAD?
   - **Editability** — can BricsCAD edit the entity; if so, how?
   - **Constructability** — does BricsCAD have a command for creating the entity?
4. We made a screen grab of each entity in AutoCAD and then following translation in BricsCAD. The before and after images are included in this chapter illustrate similarities and differences.
5. We made a record the limitations we found.

The results of these tests are presented on the following pages.

Decoding the Legend

In this chapter, we mark how well BricsCAD supports each AutoCAD entity by means of this legend:

Entity Name	*READ / CREATE / EDIT*

The words in the legend have the following meaning.

READ — BricsCAD reads the entity from DWG files, and displays it correctly

CREATE — BricsCAD can create the entity

EDIT — BricsCAD can edit the entity

There are a few AutoCAD entities that BricsCAD does not handle 100% correctly. BricsCAD can read and display dynamic blocks, but it cannot create or edit them. In these cases, the chapter tags these kinds of entities with a version of the read-edit legend that looks like this:

Dynamic Blocks	AutoCAD	BricsCAD *	*READ* / — / —

*) The footnote details the limitation

The dashes (—) in "READ / — / —" mean that BricsCAD cannot edit or create dynamic blocks, and so the words "CREATE" and "EDIT" are missing from the legend. The asterisk (*) provides additional information in the footnote on how BricsCAD handles the entity.

Summary of Problem Entities

Even though BricsCAD V17 does a very good job handling DWG files, there are some entities created by AutoCAD that are a difficulty. Here is our summary of the entities with which BricsCAD has problems.

3D Meshes

BricsCAD opens and displays 3D mesh objects created by AutoCAD's commands like Mesh and MeshSmooth, but it cannot create or manipulate them directly. They can be edited only with basic commands (such as Move, Copy, and Delete), and their basic properties can be modified, such as color and linetype.

Note that these are "true" point-based 3D mesh objects introduced to AutoCAD 2010, and not "old" meshes made from polyfaces. BricsCAD creates polyface meshes with commands like Ai_Box.

Constraints

BricsCAD has its own constraints engine, and so does not display dimensional constraints in drawings created by AutoCAD and its D-Cubed constraint engine. Geometric constraints from imported DWG files are, however, displayed.

Dimensions

Broken Dimensions. BricsCAD displays broken dimensions made by AutoCAD's DimBreak command, but cannot create or edit them.

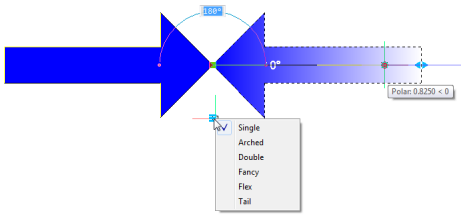
Inspection Dimensions. BricsCAD displays inspection dimensions made by AutoCAD's DimInspect command, but cannot edit or create them.

Jogged Dimensions. BricsCAD displays and edits jogged dimensions made by AutoCAD's DimJogged command, but cannot create them. BricsCAD supports the **DimJogAng** variable.

Quick Dimensioning. Bricsys initially added the QDim command to BricsCAD V15, but then pulled it after a patent licensing firm launched law suits; the quick dimensioning capabilities were allegedly patented by Adra Systems. Autodesk has since changed the function of the old Dim command to act like the dimensioning in the Quad cursor of Bricsys.

Dynamic Blocks

BricsCAD displays and edits dynamic blocks made in AutoCAD's Block Editor, but cannot create them. BricsCAD changes the look of dynamic blocks through custom grips and the Properties palette.



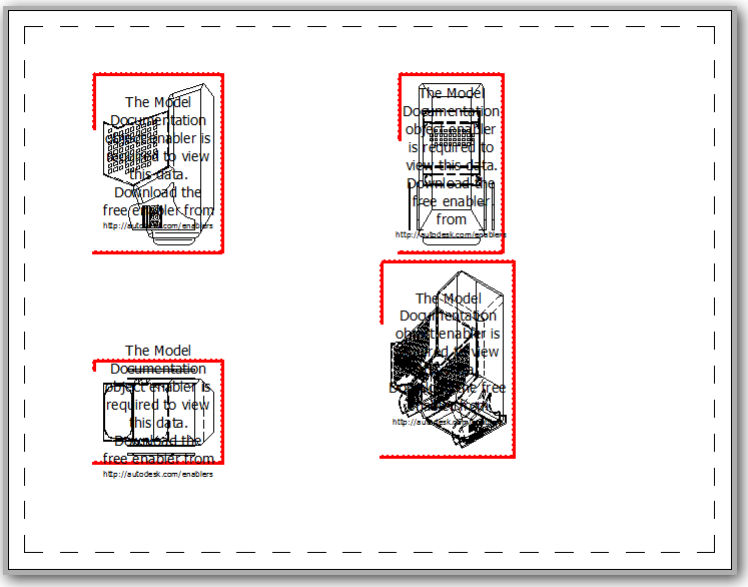
BricsCAD editing dynamic blocks through grips

Geographic Location

BricsCAD specifies geographic locations with its GeographicLocation command, but does not display, create, or edit the marker glyphs that mark locations placed in AutoCAD.

Model Documentation

BricsCAD supports model documentation created by AutoCAD's ViewBase command. The bounding boxes are displayed with a preview image of each view, but each view is also filled with a message stating a missing object enabler is needed; BricsCAD does not, however, support AutoCAD's model documentation object enabler.



Message that appears when a DWG containing AutoCAD's model documentation is opened in BricsCAD

BricsCAD has its own form of model documentation called view generation. It operates much like AutoCAD's, creates 2D plans and isometric views of 3D models, with sections and detail views.

Multilines

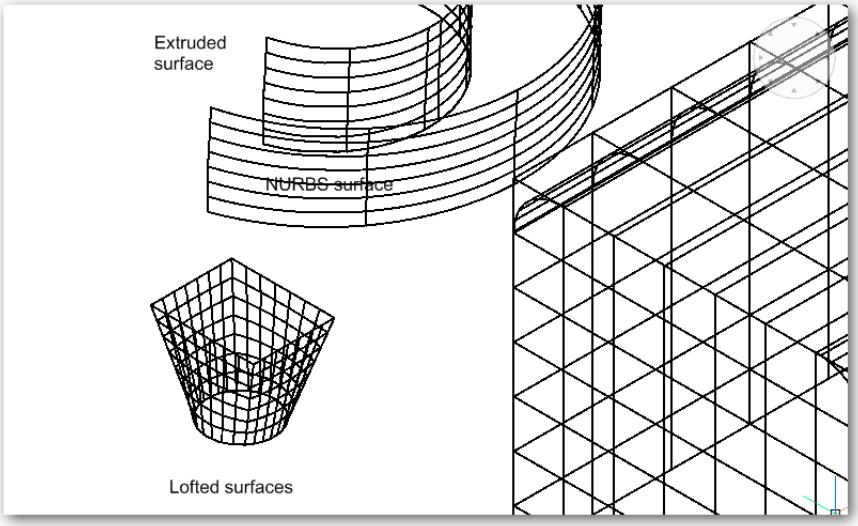
BricsCAD reads and creates multilines and multiline styles with AutoCAD's MLine and MlStyle commands. BricsCAD, however, lacks the MlEdit command, and so intersections (vertices) cannot be fully edited. Some aspects of multilines can be edited with grips and through the Properties bar's option. The BricsCAD version of the MlStyle command opens the Drawing Explorer. BricsCAD uses the same format for *.mln* multiline style files as AutoCAD, and so you can use the Drawing Explorer's **Load from MLN File** button to copy these files from AutoCAD.

Proxy Objects

BricsCAD displays proxy objects made by AutoCAD but cannot edit them, because BricsCAD does not support object enablers, except for ones for AutoCAD Architecture and Mechanical Desktop, as provided by Open Design Alliance. BricsCAD edits only the basic properties of proxy objects (color, linetype, and so on) through the Properties bar.

Surfaces

BricsCAD recognizes all surfaces created by AutoCAD, including NURBS and swept surfaces.



Surfaces created in AutoCAD and displayed by BricsCAD

Tables

BricsCAD can read, edit, and write tables, but does not quite have all the table and cell format options found in AutoCAD. For instance, it cannot place text at an angle in cells, and it cannot give cells double lines. For the complete list of BricsCAD's table style abilities, see the "Compatibility of Styles" section near the end of this chapter.

Underlays

BricsCAD does not load or display DGN and DWF underlays. It does, however, attach PDF and raster image underlays, as well as externally-referenced drawing (xrefs) files.

Viewports

BricsCAD creates and clips rectangular and polygonal viewports, but cannot invert viewports clipped by the VpClip command.

Visual Styles

BricsCAD reads, edits, and creates visual styles, but cannot apply all of the properties that AutoCAD can. For instance, the properties of Intersection Edges are not yet implemented. On the plus side, BricsCAD provides a longer list of default visual styles than does AutoCAD. See the complete list in the "Compatibility between Styles" section near the end of this chapter.

MISCELLANEOUS COMPATIBILITY ISSUES

There are aspects of CAD programs that are unaffected by DWG compatibility, yet are important to the end user . For example, I find the ribbon layout in AutoCAD overwhelming (in the negative sense), and the default white text on black background difficult to read. Other non-DWG issues include the following items:

- Overall user experience, and the layout of workspaces
- Spelling of command names and variables
- Additional commands and variables, or missing ones
- Extra palettes, options, right-click options, and other UI elements, or missing ones
- Manner in which grips operate
- Methods of customization and programming

HISTORY OF BRICSCAD'S DWG SUPPORT

Here are some of the important features added with recent releases.

BRICSCAD V11

- Arc length dimensions
- Modification of dynamic blocks through Properties bar
- Fields
- Partial support for geographic locations
- Lights
- PDF underlays
- Subdivision surfaces

BRICSCAD V12

- Dimensional and geometric constraints
- Live sections
- Tables

BRICSCAD V13

- Multilines
- Sheet sets
- Tool palettes

BRICSCAD V14

- Annotative property for text entities, dimensions, and so on
- Layer filters
- Multiline leaders and styles
- Section line entities
- 2D and 3D helix entities
- 3D solids made as swept entities and as sheet metal parts

BRICSCAD V15

- Editing of dynamic blocks and hatch patterns through grips
- Polysolid entities
- Formulae in tables

BRICSCAD V16

- 3D solid lofts, 3D surface creation, editing, and deformations
- Associative arrays
- Detail styles and sections styles
- Geomap images
- Transparency property for entities and layers

BRICSCAD V17

- AniPath for creating movies of 3D models
- SplinEdit for editing splines
- Extrude, Loft, Sweep, and Revolve create 3D surfaces from open objects
- XEdges creates lines and arcs from the edges of 3D solids

DWG 2013-7 Entity Support

To read, view, edit, and write DWG files, BricsCAD uses the Teigha library from Open Design Alliance. As ODA adds support for entities, Bricsys adds them to BricsCAD. BricsCAD V17 supports DWG AC1027, which includes entities generated by 2017 and earlier; Autodesk added no new entities to AutoCAD 2017.

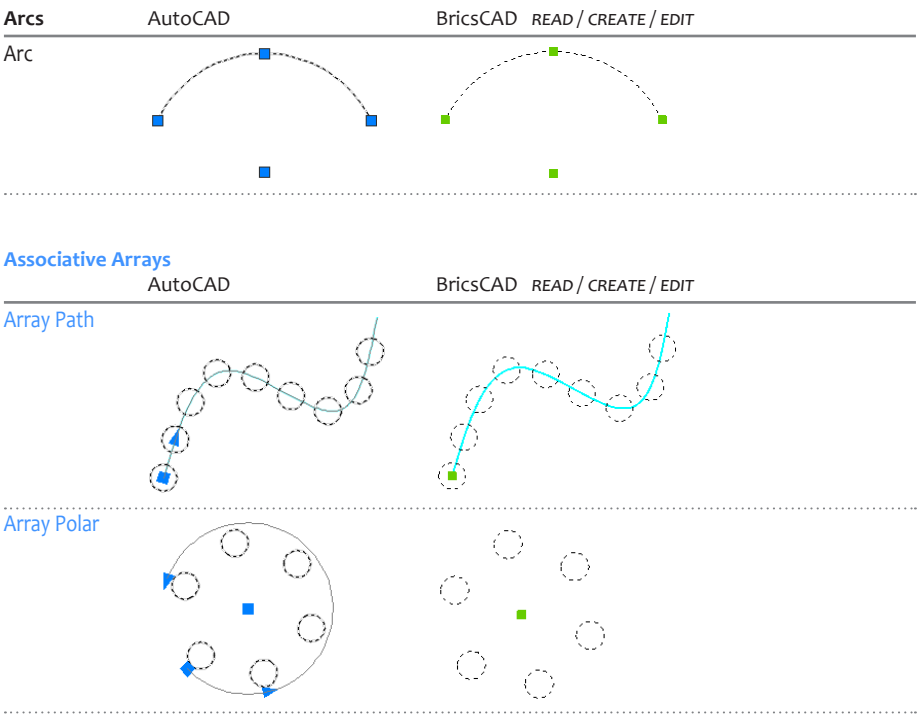
Entities are listed in alphabetical order under the following sections.

- 2D Entities
- Text Entities
- Dimension Entities
- Geometric and Dimensional Constraints
- Complex 2D Entities
- 3D Entities

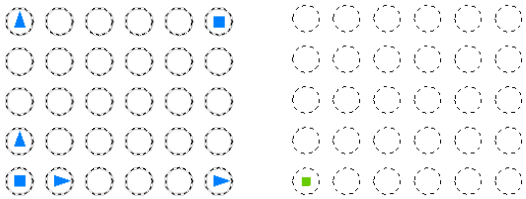
Equivalent entities are illustrated from AutoCAD and BricsCAD, with entity grips shown.

2D ENTITIES

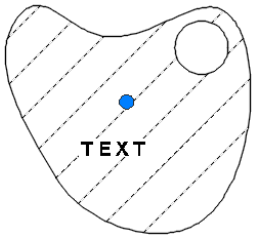
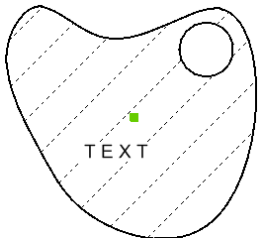
BricsCAD accurately displays the following 2D entities created in AutoCAD:

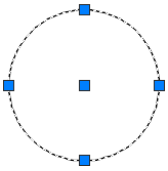
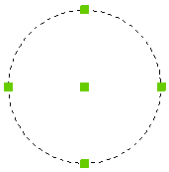


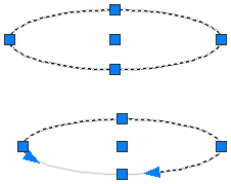
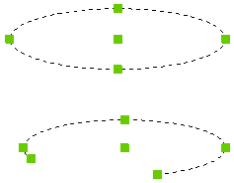
Array Rectangular

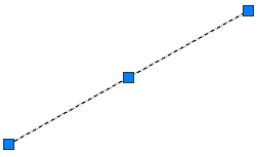
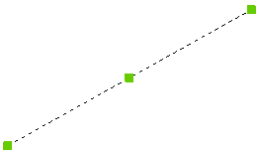


Associative Hatches

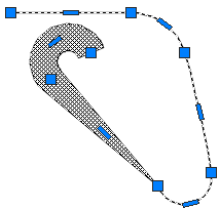
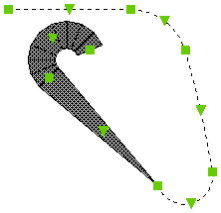
	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Hatch		

Circles	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Circle		

Ellipses	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Ellipse *Elliptical Arc*		

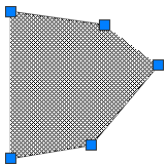
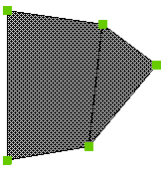
Lines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Line		

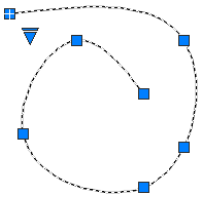
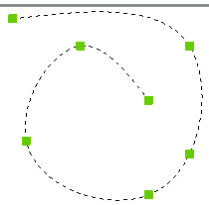
Points	AutoCAD	BricsCAD *READ / CREATE / EDIT*
PdMode 95		

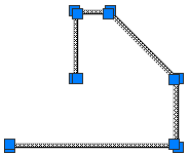
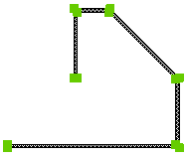
Polylines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Polyline		

Rays *	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Ray		

(*) Rays shown are cut off in this book, as real rays are infinitely long in one direction

Solids (2D)	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Solid		

Splines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Spline		

Traces	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Tracewid 50		

Xlines*	AutoCAD	BricsCAD
Xline		

(*) Xlines shown are cut off in this book, as real xlines are infinitely long in both directions

TEXT ENTITIES

BricsCAD accurately displays the following text entities created in AutoCAD. The exceptions is some formatting of mtext and tables, as detailed later in this chapter.

Attribute Definitions

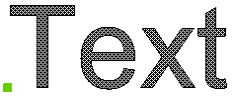
	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Attribute Definition		

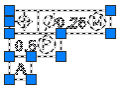
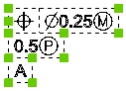
Attribute References

	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Attribute Reference		

MText	AutoCAD	BricsCAD * *READ / CREATE / PARTIAL EDIT*
Mtext		

*) BricsCAD does not create all aspects of mtext; see "Compatibility of Styles" later in the chapter.

Text	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Text		

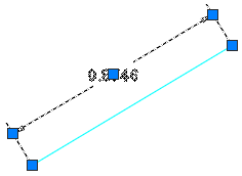
Tolerances	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Tolerance		

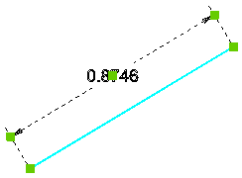
DIMENSION ENTITIES

BricsCAD supports all aspects of AutoCAD's dimension entities, except that it cannot create or edit broken, inspection, and jogged dimensions.

Aligned **AutoCAD** **BricsCAD** *READ / CREATE / EDIT*

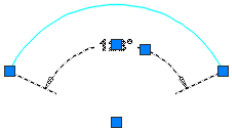
Aligned Dimension

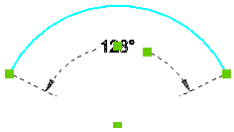




Angular **AutoCAD** **BricsCAD** *READ / CREATE / EDIT*

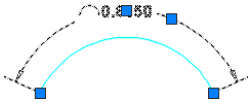
Angular Dimension

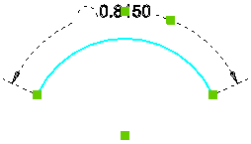




Arc Length **AutoCAD** **BricsCAD *** *READ / — / —*

Arc Length Dimension

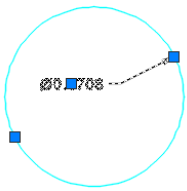


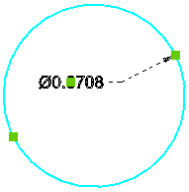


*) BricsCAD displays arc length dimensions, but does not create or edit them.

Diameter **AutoCAD** **BricsCAD** *READ / CREATE / EDIT*

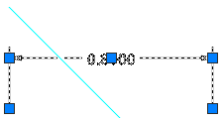
Diametric Dimension

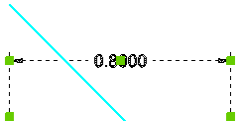




DimBreak **AutoCAD** **BricsCAD *** *READ / — / —*

Broken Dimension Line

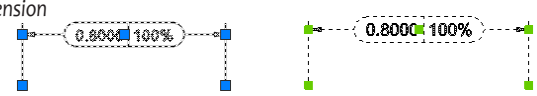




*) BricsCAD displays broken dimensions, but does not create or edit them.

DimInspect	AutoCAD	BricsCAD *	READ / — / —

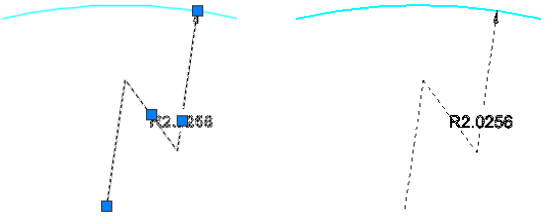
Inspection Dimension



*) BricsCAD displays inspection dimensions, but does not create or edit them.

DimJogged	AutoCAD	BricsCAD *	READ / — / EDIT

Jogged Dimension



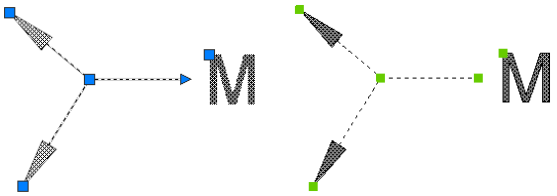
*) BricsCAD displays and edits jogged dimensions, but does not create them.

Leaders	AutoCAD	BricsCAD	READ / CREATE / EDIT

Leader

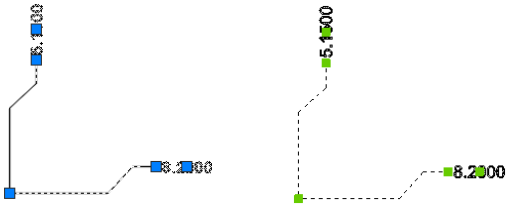
Multileaders	AutoCAD	BricsCAD	READ / CREATE / EDIT

Multiline Leader



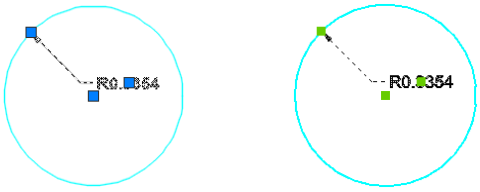
Ordinate	AutoCAD	BricsCAD	READ / CREATE / EDIT

Ordinate



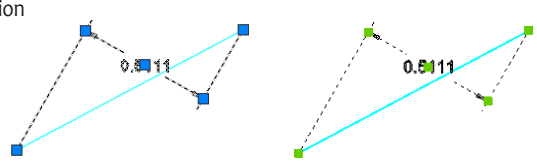
Radius	AutoCAD	BricsCAD	READ / CREATE / EDIT

Radial Dimension



Rotated **AutoCAD** **BricsCAD** *READ / CREATE / EDIT*

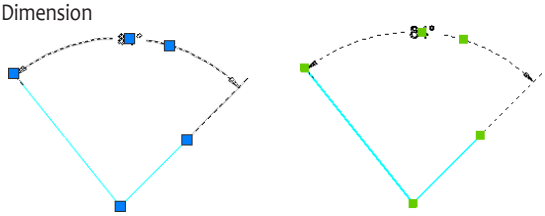
Rotated Dimension



Three-Point Angular

AutoCAD **BricsCAD** *READ / CREATE / EDIT*

3-point Angular Dimension

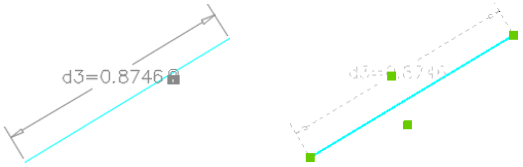


GEOMETRIC AND DIMENSIONAL CONSTRAINTS

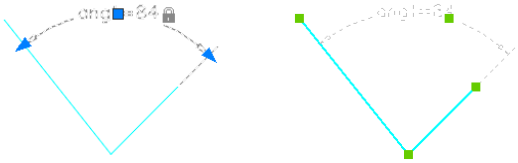
Dimensional Constraints

AutoCAD BricsCAD *READ / CREATE / EDIT*

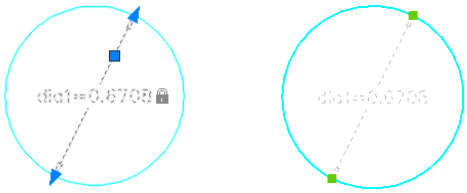
Aligned Dimensional Constraint



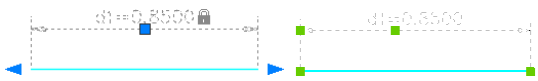
Angular Dimensional Constraint



Diameter Dimensional Constraint

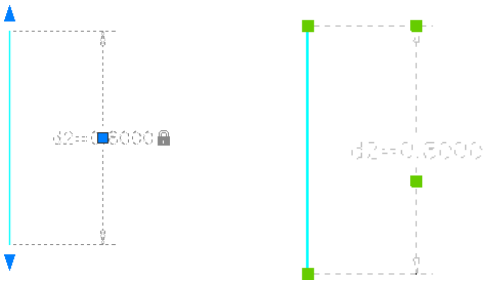


Horizontal Dimensional Constraint

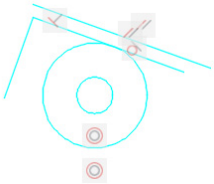
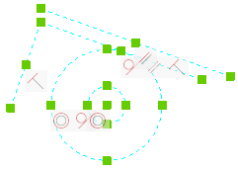


Radius Dimensional Constraint

Vertical Dimensional Constraint

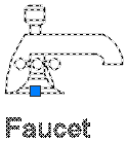
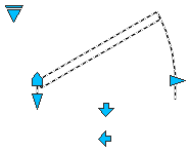
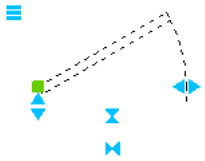


Geometric Constraints

	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Geometric constraint		

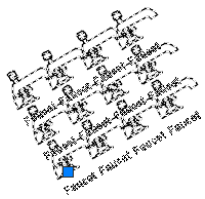
COMPLEX 2D ENTITIES

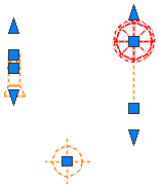
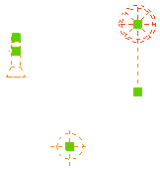
Block References

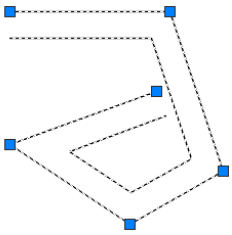
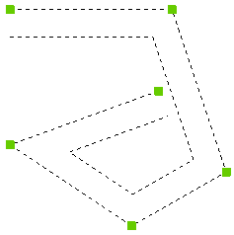
	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Insert		
*Dynamic **		

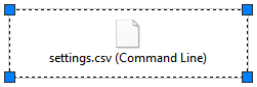
*) BricsCAD displays and edits dynamic blocks, but does not create them.

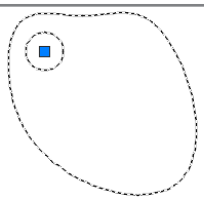
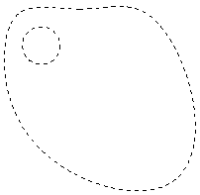
MInsert

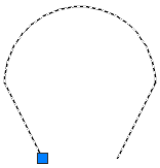
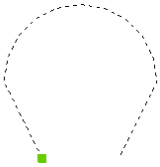


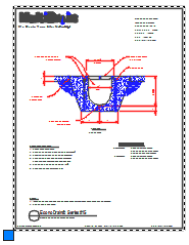
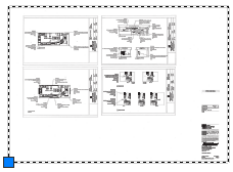
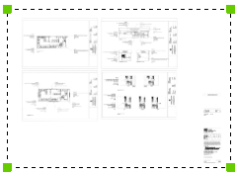
Lights	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Light		

Multilines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Mline		

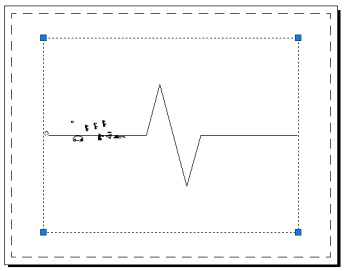
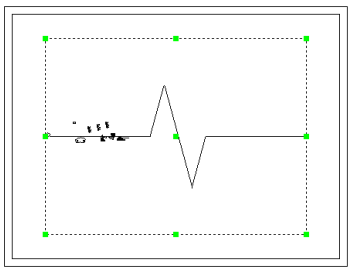
OLE Frames	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Ole		

Regions	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Region		

Shapes	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Shape		

Underlays	AutoCAD	BricsCAD * *PARTIAL READ / CREATE / EDIT*
DWF Underlay		*) BricsCAD does not display DWF underlays.
PDF Underlay		

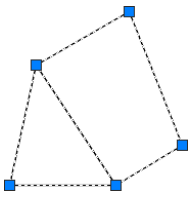
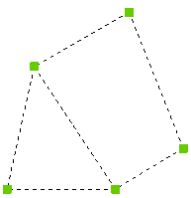
*) Bricsys does not support multi-page PDF files.

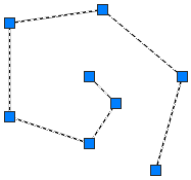
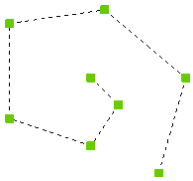
Viewports	AutoCAD	BricsCAD * *READ / CREATE / PARTIAL EDIT*
Viewport		

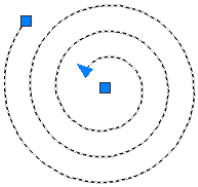
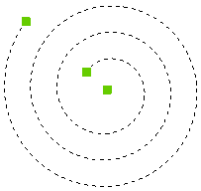
*) BricsCAD does not invert clipped viewports.

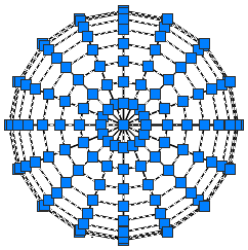
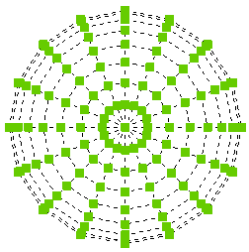
3D ENTITIES

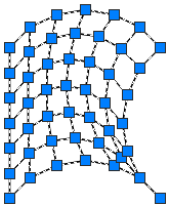
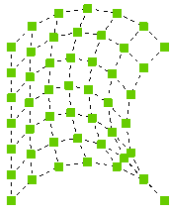
BricsCAD accurately displays the following 3D entities created in AutoCAD:

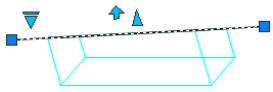
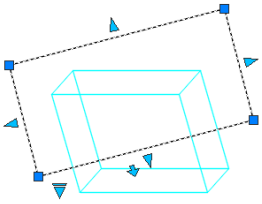
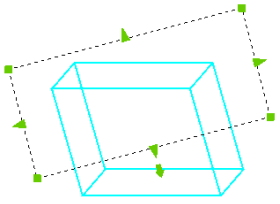
3D Faces	AutoCAD	BricsCAD *READ / CREATE / EDIT*
3D face		

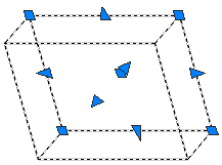
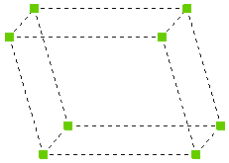
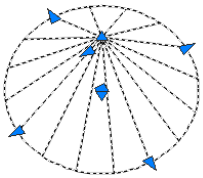
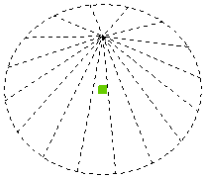
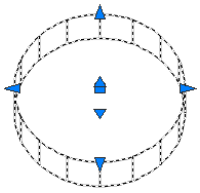
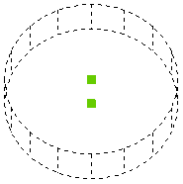
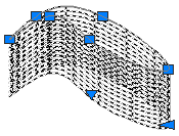
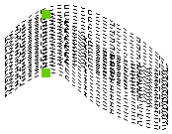
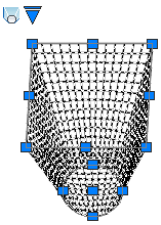
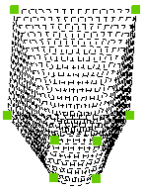
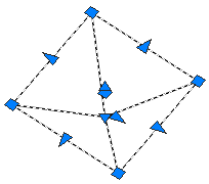
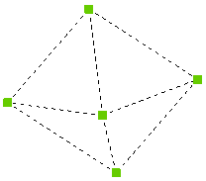
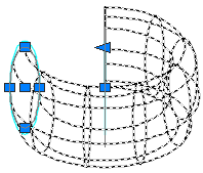
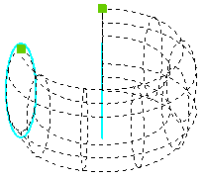
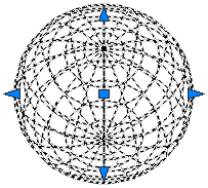
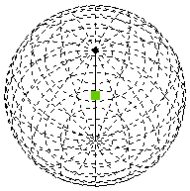
3D Polylines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
3D polyline		

Helixes	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Helix		

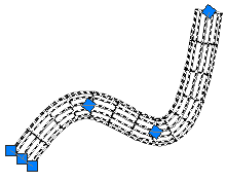
Polyface Meshes	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Polyface Mesh		

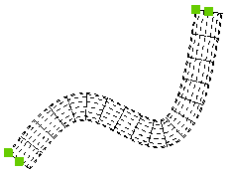
Polygon Meshes	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Polygon Mesh		

Sections	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Section Line		
Section Object		

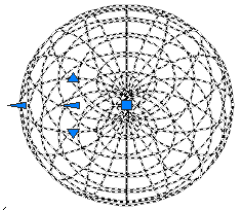
3D Solids	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Box		
Cone		
Cylinder		
Extrusion		
Loft		
Pyramid		
Revolve		
Sphere		

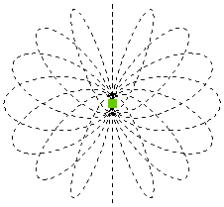
Sweep



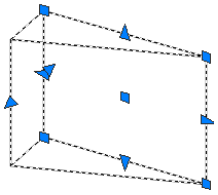


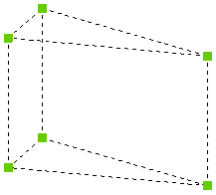
Torus





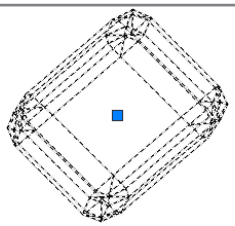
Wedge

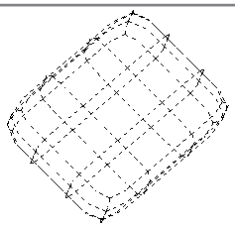




Subdivisions	AutoCAD	BricsCAD *	*READ* / — / —

Mesh

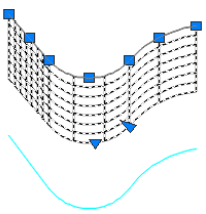


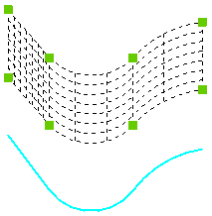


*) BricsCAD recognizes mesh objects created by AutoCAD, but cannot create or manipulate 3D meshes. The objects can be edited using basic commands (such as Move, Copy, and Delete), and their basic properties can be modified, such as color and linetype. (Note that these are the "true" 3D mesh objects introduced recently to AutoCAD, and not the "old" meshes made from polyfaces like Ai_Box and Ai_Sphere.)

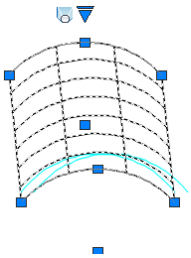
Surfaces	AutoCAD	BricsCAD *	*READ* / *CREATE* / *EDIT*

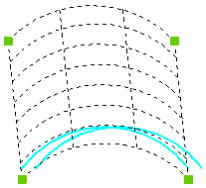
Extrusion





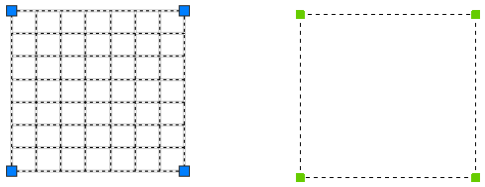
Loft



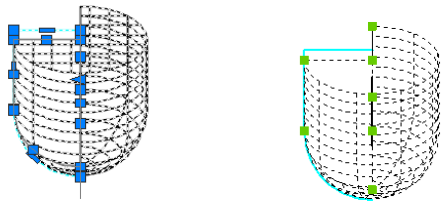


NURBS

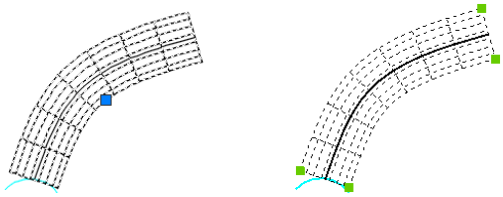
Planar



Revolve



Sweep



Compatibility Between Properties

BricsCAD supports most of the entity properties found AutoCAD, including the BYLAYER and BYBLOCK settings.

AutoCAD Property	BricsCAD Property	Notes
Annotative	Annotative	
Color	Color	BricsCAD supports ACI colors and True Colors, but not color books
Elevation	Elevation	
Hyperlink	Hyperlink	
Layer	Layer	BricsCAD supports all layer names
Linetype	Linetype	BricsCAD supports all AutoCAD linetypes, and reads *.lin* files
Linetype scale	Linetype Scale	
Lineweight	Lineweight	BricsCAD supports all lineweights
Material	Material	BricsCAD has its own materials library
Plot Style	Plot Style	BricsCAD supports AutoCAD plot styles, reads *.ctb* and *.stb* files
Shadow display	...	BricsCAD does not support shadow property
Thickness	Thickness	
Transparency	Transparency	

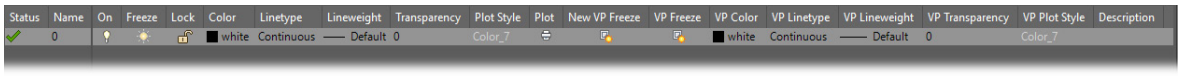
LAYER PROPERTY COMPATIBILITY

BricsCAD supports all of the basic properties of AutoCAD's layering system. For instance, DWG files can contain an unlimited number of layers, with names up to 255 characters long, including special characters.

BricsCAD supports layer states and filters, like AutoCAD; on the other hand, BricsCAD supports the Material property in directly layers, whereas AutoCAD does only indirectly.

AutoCAD Command	BricsCAD Command	Comment
Layer	LayerPanelOpen	Opens the Layer panel (palette)
LayerCLose	LayerPanelClose	Closes the Layer panel
ClassicLayer	Layer	Opens the Layer dialog box
LayerState	LayerState	
LayerP	LayerP	

The figures below illustrate the differences between the layer properties in both CAD system:

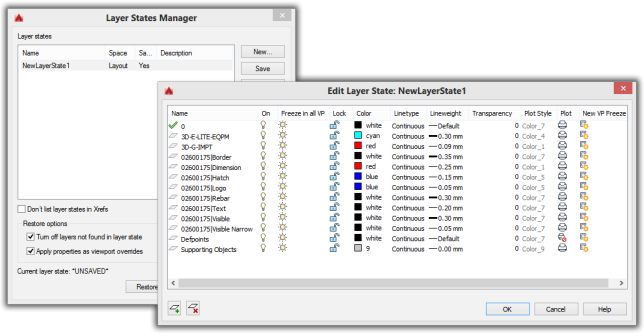


Above: *Layer properties in AutoCAD (paper space)*
Below: *Layer properties in BricsCAD (paper space)*

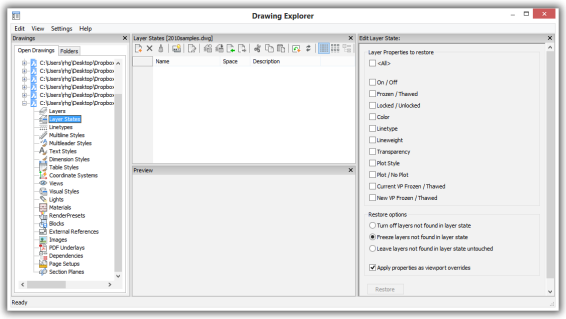
The differences in layer properties are listed concisely by the following table:

AutoCAD Layer Property	BricsCAD Equivalent Property	Notes
Status	Current	BricsCAD supports two statuses: current or not current
Name	Layer Name	BricsCAD supports all AutoCAD forms of layer names
On	On/Off	
Freeze	Freeze	
Lock	Locked	
Color	Color	BricsCAD supports all AutoCAD colors, except ColorBooks
Linetype	Linetype	BricsCAD supports all AutoCAD linetypes, and the *.lin* file
Lineweight	Lineweight	
Transparency	Transparency	
Plot Style	Plot Style	BricsCAD supports AutoCAD plot styles formats, *.ctb* and *.stb* files
Plot	Plot	
New VP Freeze	New VP	
VP Freeze	VP Freeze	
VP Color	VP Color	
VP Linetype	VP Linetype	
VP Lineweight	VP Lineweight	
VP Transparency	VP Transparency	
VP Plot Style	VP Plot Style	
Description	Description	
...	Material	BricsCAD assigns materials to 3D objects though layers

BricsCAD defines and controls layer states through its ubiquitous Drawing Explorer.



Left: *Layer States Manager dialog boxes in AutoCAD*
Right: *Layer States in BricsCAD's Drawing Explorer*



Compatibility Between Styles

BricsCAD supports most of the styles found AutoCAD.

AutoCAD Style	BricsCAD Style	Notes
Detail view styles	Detail view styles	
Dimension styles	Dimension styles	
Leader, QLeader	DimLeader, QLeader	
Multiline styles	Multiline styles	BricsCAD supports all aspects of multilines, except editing intersections
Multileader styles	Multileader styles	
Plot styles	Plot styles	
Section view styles	Section view styles	
Section styles	Section Planes	
Table styles	Table styles	BricsCAD supports most aspects of table styles, and annotative scaling
Text styles	Text styles	BricsCAD supports most aspects of text styles, and annotative scaling
Visual styles	Visual styles	BricsCAD supports many aspects of visual styles

The following sections describe style compatibility in greater detail.

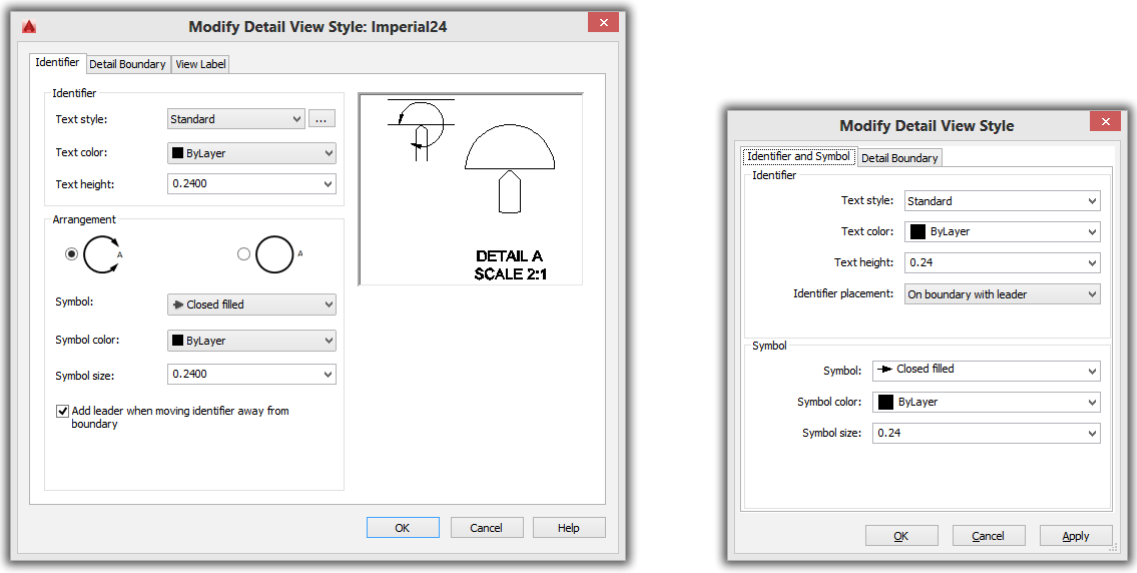
View Detail and Section Styles

Detail and section view styles are part of AutoCAD's model documentation function. In BricsCAD, this documentation is called drawing views (formerly known as "generative drafting").

Model documentation and *drawing views* are the CAD system's ability to make traditional 2D views — front, right, top, isometric, and so on — from 3D models automatically. In AutoCAD, they can be sourced from AutoCAD or Inventor. Both CAD systems work with models imported from other MCAD systems such as Solidworks and Pro/Engineer.

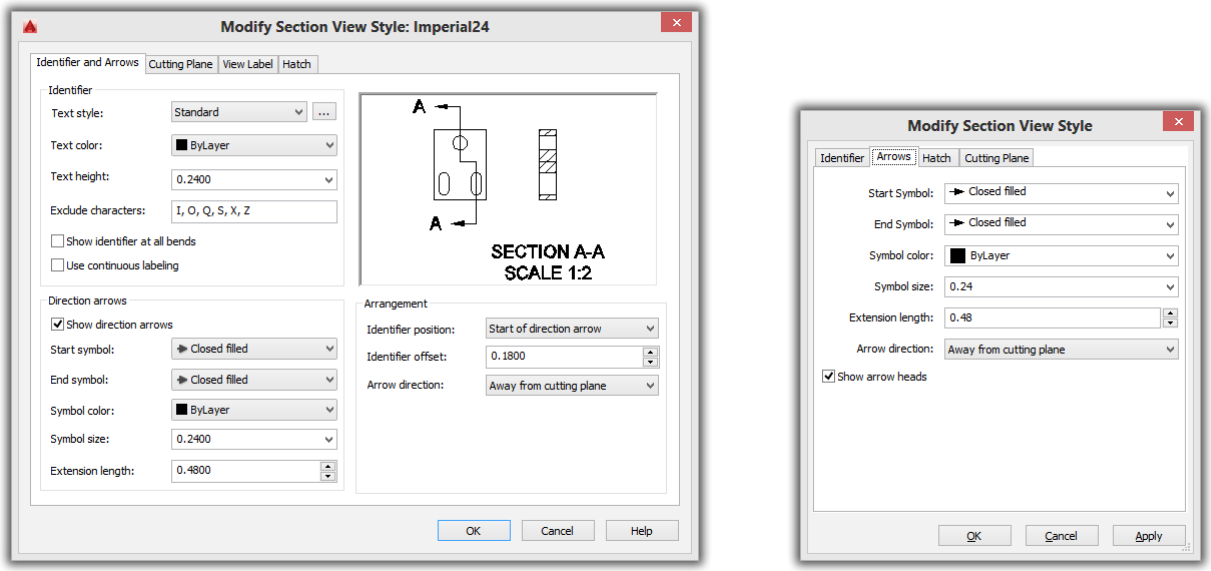
AutoCAD Commands	BricsCAD Commands
ViewDetailStyle	ViewDetailStyle
ViewSectionStyle	ViewSectionStyle

The ViewDetailStyle and ViewSectionStyle commands are new to BricsCAD, and so it does not support as many style aspects as does AutoCAD. Here are the dialog boxes displayed by the **ViewDetailStyle** command:



Left: *AutoCAD's tabbed Modify Detail View Style dialog box;* **right:** *BricsCAD's tabbed Modify Detail View Style dialog box*

Dialog boxes from AutoCAD and BricsCAD for the **ViewSectionStyle** command:



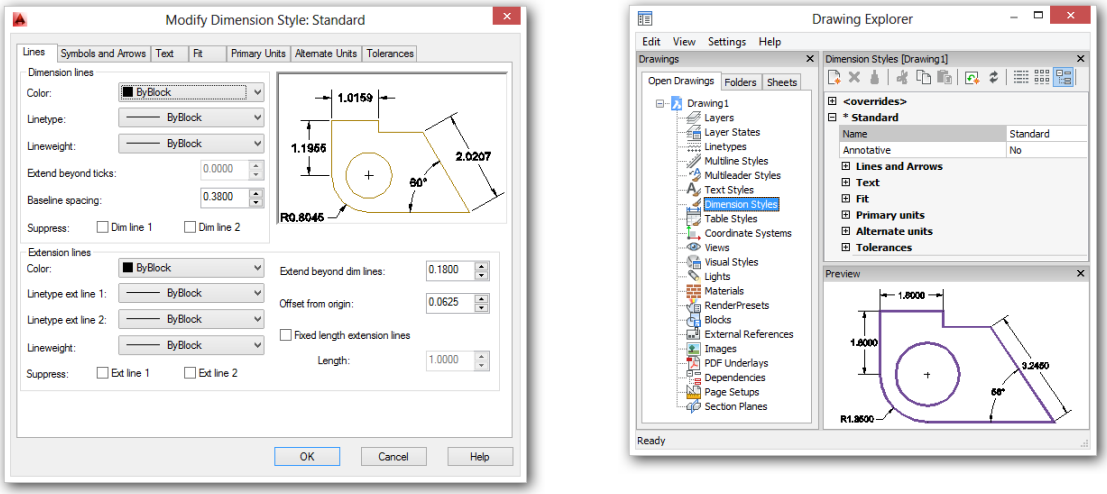
Left: *AutoCAD's tabbed Modify Section View Style dialog box;* **right:** *BricsCAD's tabbed Modify Section View Style dialog box*

Dimension Styles

BricsCAD supports all properties of AutoCAD's dimension styles and variables, with the exception of text direction.

AutoCAD Command	BricsCAD Command
DimStyle	DimStyle

In BricsCAD, the DimStyle command brings up the Drawing Explorer:



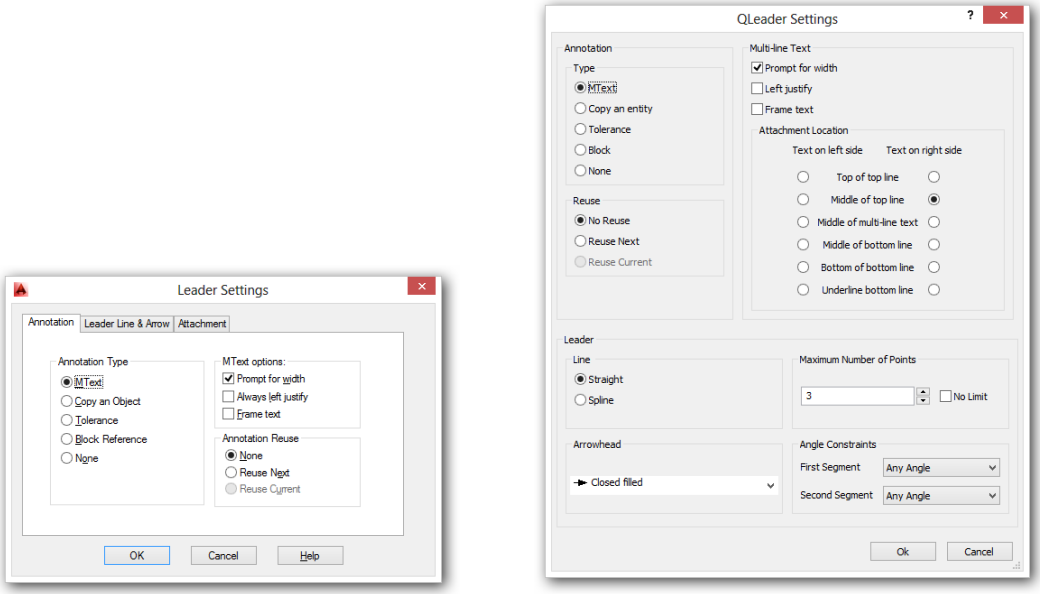
Left: *AutoCAD's DimStyle tabbed dialog box;* **right:** *BricsCAD's Drawing Explorer for dimensions styles*

Leader and QLeader Styles

BricsCAD supports styles for leaders (drawn by the DimLeader or QLeader commands) through the DimStyle command, just like AutoCAD. V14 added support for multiline leaders; see the later section.

AutoCAD Commands	BricsCAD Commands
Leader, DimStyle	DimLeader, DimStyle
QLeader, QLeader Setting	QLeader, QLeader Setting

Unlike most other styles, the options for QLeader in BricsCAD are accessed through a dialog box via the QLeader command's Settings option.



Left: *AutoCAD's QLeader command's Settings dialog box;* **right:** *BricsCAD's QLeader command's options.*

BricsCAD supports all the QLeader options found in AutoCAD.

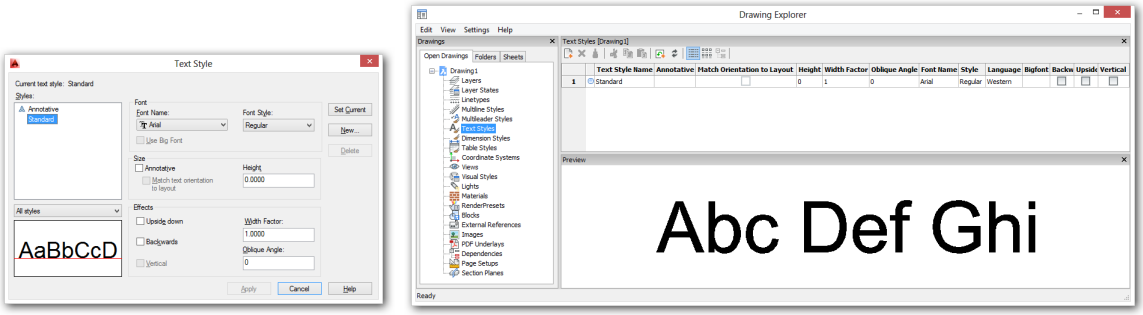
AutoCAD QLeader Option	Equivalent BricsCAD Option
Annotation options	
Annotation Type	Type
MText Options	Multi-line Text
Annotation Reuse	Reuse
Leader Line & Arrow options	
Leader Line	Leader
Number of Points	Maximum Number of Points
Arrowhead	Arrowhead
Angle Constraints	Angle Constraints
Attachment options	
Text on left side	Text on left side
Text on right side	Text on right side
Underline bottom line	Underline bottom line

MText and Text Styles

BricsCAD supports all of AutoCAD's text style options. BricsCAD uses an icon for annotative text styles that looks somewhat different from AutoCAD's:

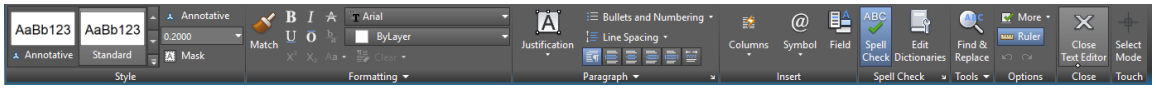
Left: *Annotation icon used by AutoCAD;* ***right:*** *As employed by BricsCAD*

AutoCAD Command	BricsCAD Command
Style	Style
MText	MText

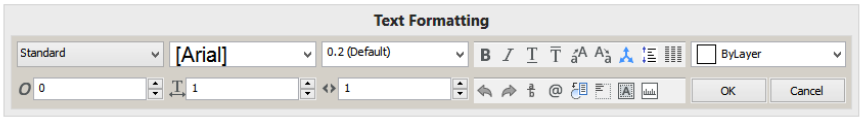


Left: *AutoCAD's Style dialog box;* **right:** *BricsCAD's Drawing Explorer for text styles.*

The MText toolbars for both CAD systems are shown below.



Above: *AutoCAD's mtext editing ribbon;* **below:** *BricsCAD's mtext editing toolbar*



BricsCAD supports most of AutoCAD's mtext options, including mtext's ability to override styles.

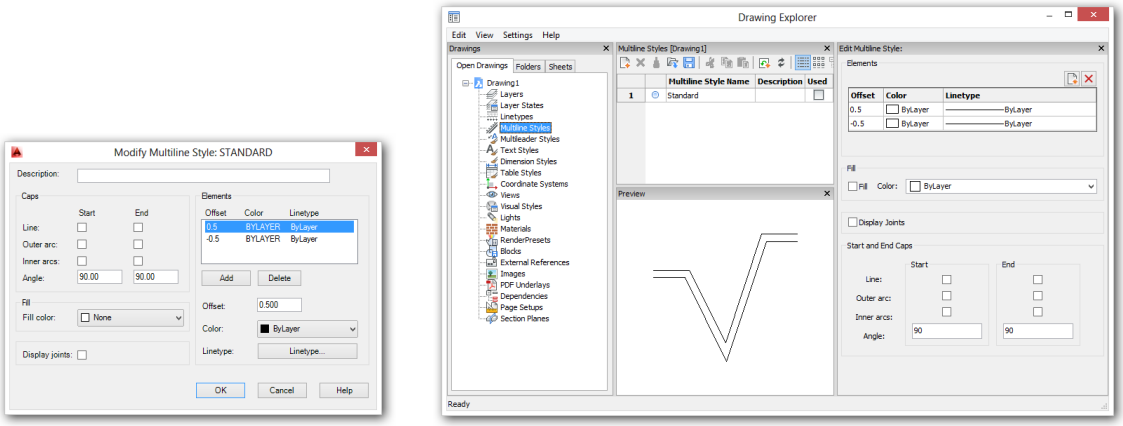
AutoCAD Mtext Function	BricsCAD Mtext Function
Style	Style
Font	Font
Annotative	Annotative
Height	Height
Boldface	Boldface
Italicized	Italicized
Underline	Underline
Overline	Overline
Undo	Undo
Redo	Redo
Fractions	Fractions
Color	Color
Ruler Toggle	Ruler Toggle
Dynamic or Static Columns	Dynamic or Static Columns
Column Properties	Column Properties
Text Justification	Text Justification
Paragraph Properties	...
Paragraph Justification	Paragraph Justification
Line Spacing	Line Spacing
Bullets	...
Field Text	Field Text
Case Conversion	Case Conversion
Special Characters	Special Characters
Obliquing Angle	Obliquing Angle
Tracking	Tracking
Width Factor	Width Factor
Import Text	*(Use PasteSpec command)*
Find and Replace	*(Use the Find command)*
AutoCAPS	...
Character Set	...
Combine Paragraphs	...
Remove Formatting	...
Background Mask	Background Mask
Editor Settings	*(Use Settings command)*

Multiline Styles

BricsCAD creates multilines through the MLine command and specifies their styles through the MlStyle command, which brings up the Drawing Explorer.

AutoCAD Command	BricsCAD Command
MlStyle	MlStyle

BricsCAD supports all properties found in AutoCAD's multiline styles.



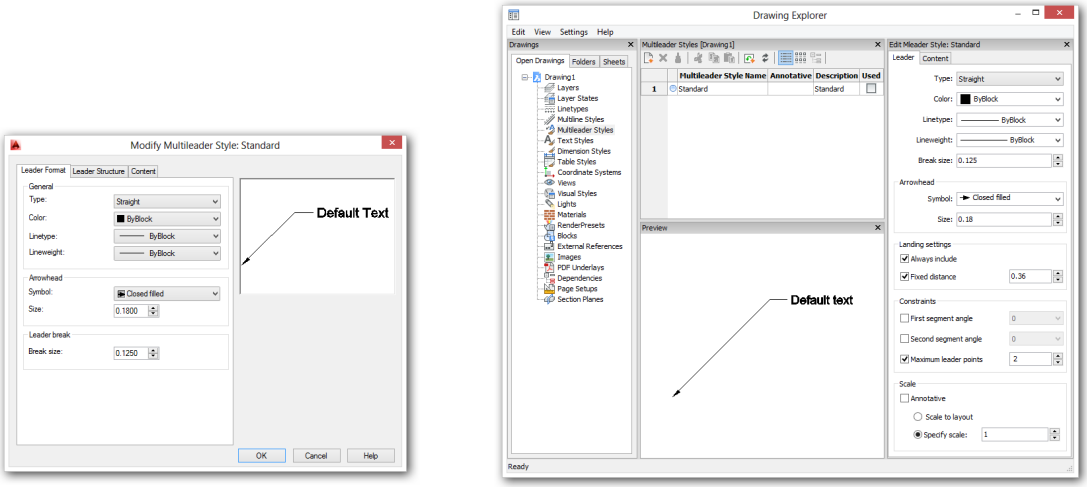
Left: *AutoCAD's multiline style editor;* ***right*** *BricsCAD's multiline style editor in Drawing Explorer*

Multileader Styles

BricsCAD creates leaders with multiple lines through the MLeader command and specifies their styles through the MleaderStyle command, which brings up the Drawing Explorer.

AutoCAD Command	BricsCAD Command
MleaderStyle	MleaderStyle

BricsCAD supports all properties found in AutoCAD's multileader style dialog box, except that it lacks the callout blocks included with AutoCAD.



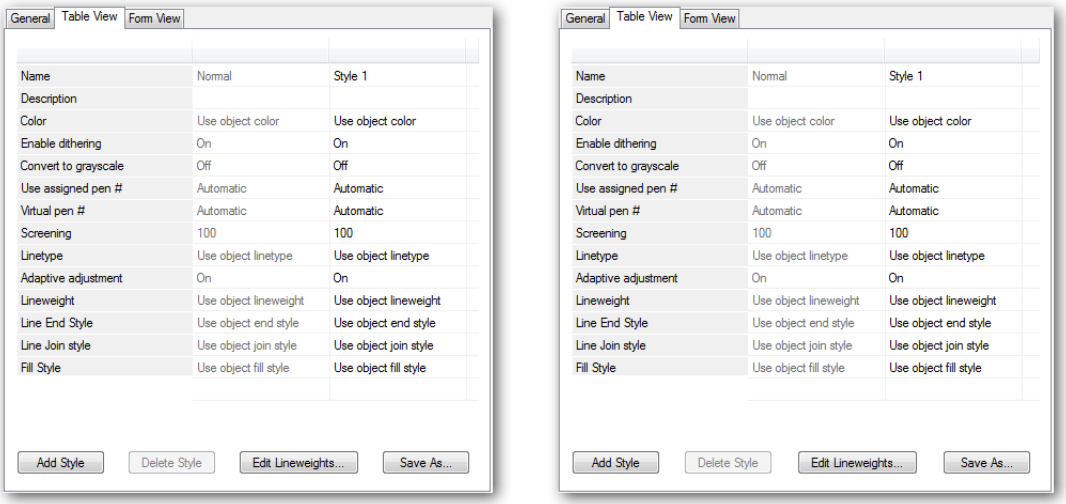
Left: *AutoCAD's multileader style editor;* ***right*** *BricsCAD's multileader style editor in Drawing Explorer*

Plot Styles

BricsCAD supports both types of AutoCAD plot styles, color and table-based. They are created and edited with the same commands as in AutoCAD.

AutoCAD Command	BricsCAD Command
PlotStyle	PlotStyle
StylesManager	StylesManager
PlotterManager	PlotterManager
PageSetup	PageSetup

The properties supported for plot styles are identical in both CAD systems — color-based styles stored in *.ctb* files; table-based styles stored in *.stb* files.



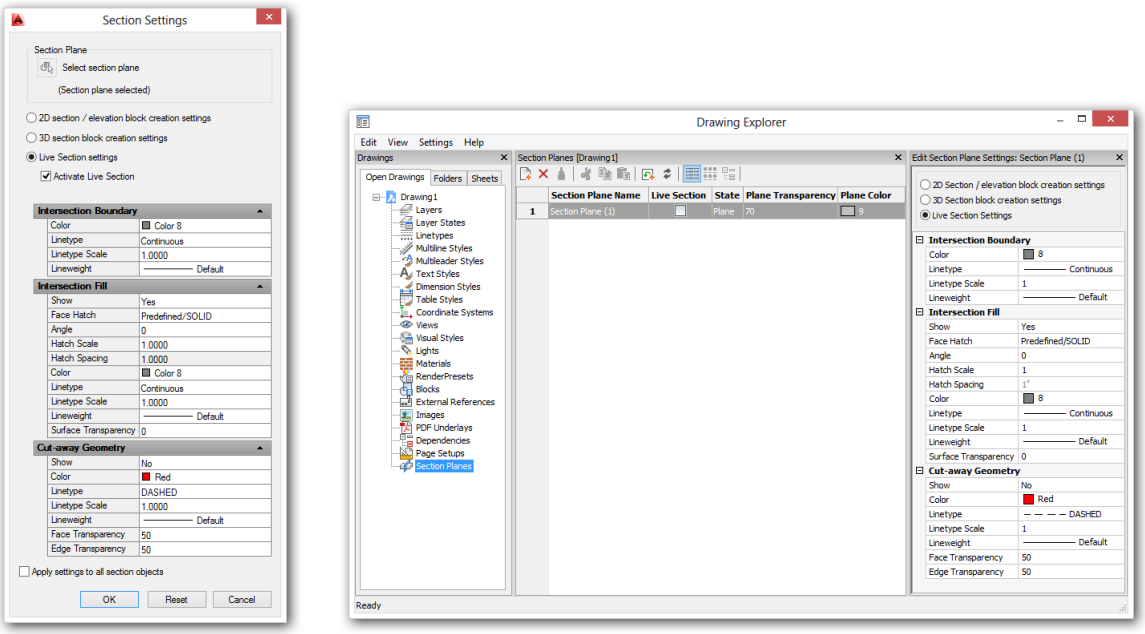
Left: *Plot style properties in AutoCAD... ;* ***right:*** *...and in BricsCAD.*

Section Styles

BricsCAD supports all the same section style properties as in AutoCAD. This includes 2D, 3D, and live sections of 3D models. Section properties are created and edited with the same commands as in AutoCAD.

AutoCAD Command	BricsCAD Command
SectionPlaneSettings	SectionPlaneSettings

Section styles are created and modified in BricsCAD by the Drawing Explorer:



Left: *Section Settings palette in AutoCAD;* ***right:*** *Section Planes settings in BricsCAD's Drawing Explorer*

Table Styles

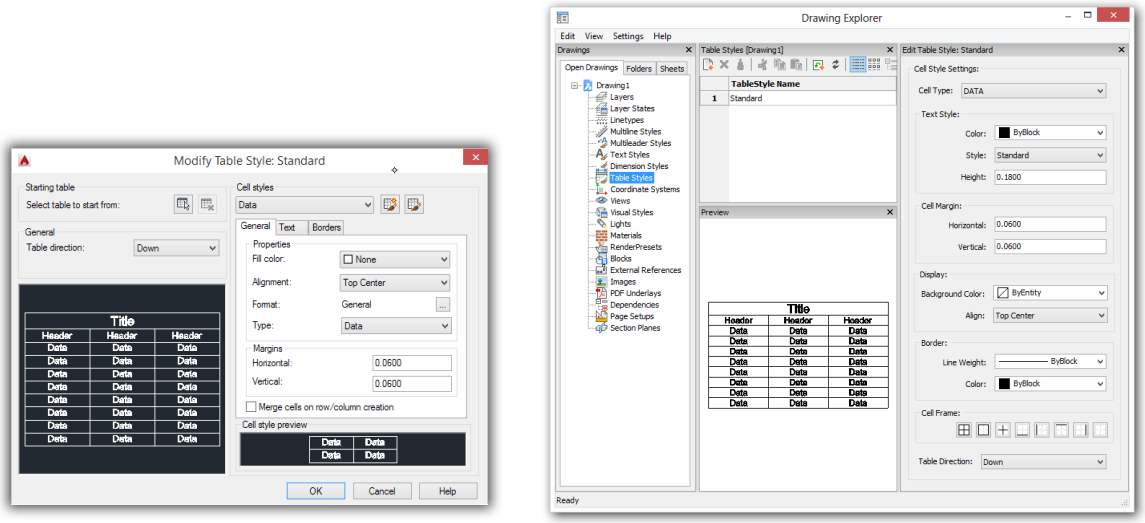
BricsCAD creates and edit table styles with the TableStyle command, as in AutoCAD. Unlike AutoCAD, BricsCAD's TableStyle command calls up the Table Style section of the ubiquitous Drawing Explorer.

AutoCAD Command	BricsCAD Command
TableStyle	TableStyle

Like AutoCAD, BricsCAD formats cells separately as "titles," "headers," and "data." BricsCAD does not support all of the table properties handled by AutoCAD, as detailed by the table below.

AutoCAD Table Property	Equivalent BricsCAD Table Property
General (Data) properties	
Table Direction	Table Direction
Fill Color	Background Color
Alignment	Align
Text Format	(see Text Properties)
Cell Margins	Cell Margins
Merge Cells	...
Text properties	
Style	Style
Height	Height
Color	Color
Angle	...

Borders properties	
Lineweight	Lineweight
Linetype	...
Color	Color
Double Line	...
Double Line Spacing	...
Apply to Borders	Cell Frame



Left: *AutoCAD's table properties edited in Modify Table Styles dialog box;* **right:** *BricsCAD's table properties edited in the Drawing Explorer*

Visual Styles

BricsCAD has all the same named visual styles as AutoCAD, plus a few extras.

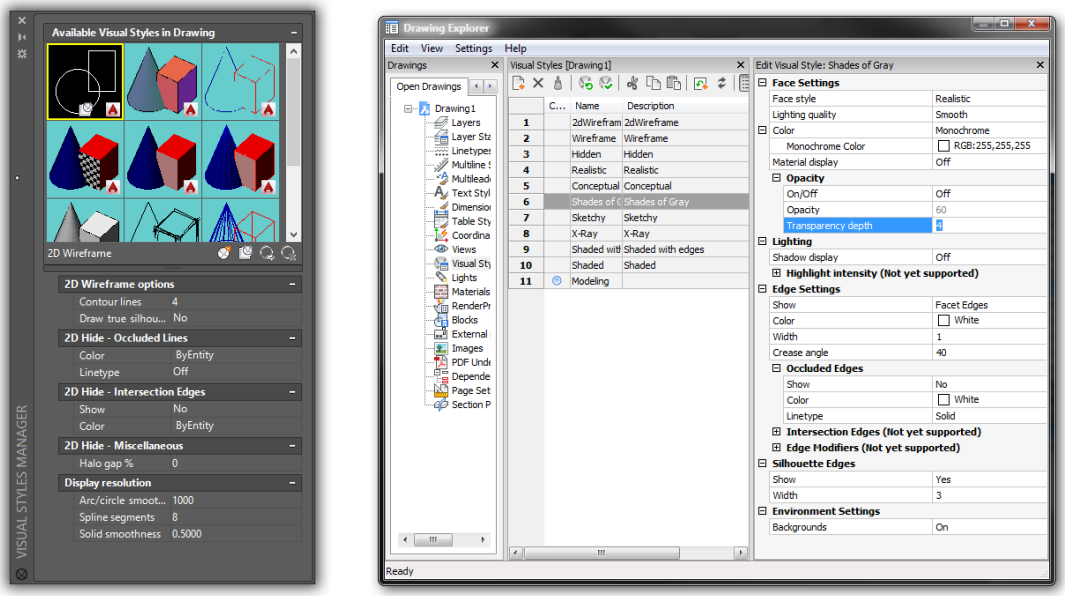
AutoCAD Command	BricsCAD Command
VsCurrent	ShadeMode
VisualStyles	VisualStyles
-VisualStyles	-VisualStyles

BricsCAD includes the following visual styles. Those shown in blue were added to BricsCAD since the last edition of this book, and some have been renamed.

AutoCAD Visual Style Name	BricsCAD Visual Style Name
2dwireframe	2dWireframe
Wireframe	Wireframe (formerly 3D Wireframe)
Hidden	Hidden (formerly 3D Hidden)
Realistic	Realistic
Conceptual	Conceptual
...	Modeling

Shaded	Shaded (formerly Gouraud)
shaded with Edges	Shaded with Edges
shades of Gray	Shades of Gray
SKetchy	Sketchy
X-ray	X-Ray

Custom visual styles cannot be exported or imported from or to both CAD packages. BricsCAD's VisualStyles command opens Drawing Explorer for creating and editing visual styles:



Left: *Visual Styles Manager in AutoCAD.*
Right: *Drawing Explorer for editing visual styles in BricsCAD.*

BricsCAD supports most of AutoCAD's visual style properties, but has some that are missing from AutoCAD.

AutoCAD Visual Style Property	Equivalent BricsCAD Property
Face Settings properties	
Face Style	Face Style
Lighting Quality	Lighting Quality
Color	Color
Monochrome Color	Monochrome Color
Material Display	Material Display
Opacity properties	
...	On/Off
...	Opacity
...	Transparency Depth
Lighting properties	
Highlight Intensity	...
Shadow Display	Shadow Display

Environmental Settings properties	
Backgrounds	Backgrounds
Edge Settings properties	
Show	Show
Color	Color
...	Width
...	Crease Angle
Occluded Edges properties	
Show	*Show*
Color	*Color*
Linetype	*Linetype*
Silhouette Edges properties	
Show	*Show*
Width	Width
Intersection Edges properties	*(not yet implemented)*
Show	...
Color	...
Linetype	...
Edge Modifiers properties	*(not yet implemented)*
Extension Lines	...
JItter	...
Crease Angle	...
Halo Gap%	...

This chapter showed how well BricsCAD reads, creates, and edits nearly the same entities as AutoCAD. Compatibility is important enough for Bricsys to improve the capabilities of BricsCAD with each release.

CHAPTER FOUR

Customizing and Programming BricsCAD

FOR END USERS WISHING TO CUSTOMIZE BRICSCAD OR AUTOCAD, MOST OF THE ACTIVITY takes place inside dialog boxes accessed by these commands:

BricsCAD: through the **Settings** (alias: options) and **Customize** (alias: cui) commands
AutoCAD: through the **Options** and **Cui** commands

The **Settings** command in BricsCAD (Options in AutoCAD) configures the way the CAD program looks and operates, while the **Customize** (Cui in AutoCAD) command changes the actions of user interface elements, such as menus, ribbon, and mouse buttons. The programming of add-ons takes place through built-in languages, such as LISP and VBA or through external programming links like BRX (ARx in AutoCAD) and .Net.

This chapter provides you with an overview of customizing and programming BricsCAD. Its emphasis is on the way that BricsCAD does things differently from AutoCAD; there is, after all, no need to learn what's the same!

Additional information is available from these sources:

- For complete details on these topics, see the *Customizing BricsCAD* ebook, available for purchase from https://www.bricsys.com/estore
- For detailed information on programming BricsCAD, refer to the online developer reference available free at https://www.bricsys.com/bricscad/help/en_US/V17/DevRef

CUSTOMIZATION CAPABILITIES

This table illustrates the similarity in customization capabilities between AutoCAD and BricsCAD. Customization methods discussed in this chapter are shown in **boldface**.

Area of Customization	AutoCAD Command	Equivalent Command in BricsCAD
Aliases	**...**[1]	**Customize | Aliases**
Command bar	Options | Display	Settings | Command Line
Cursor	Options | Display	Settings | Display
Double-click actions	**Cui | Double-click Actions**	**Customize | Mouse**
Dynamic input	Options | Drafting	Settings | Dynamic Input
File paths	**Options | Files**	**Settings | Files**
Fonts	**Style**	**Style**
Grips	Options | Selection	Settings | Grips
Hatch patterns	**...**[1]	**...**[1]
Keyboard shortcuts	**Cui | Keyboard Shortcuts**	**Customize | Keyboard**
Linetypes	**...**[1]	**Explorer** [1]
Menu bar	**Cui | Menus**	**Customize | Menu**
Mouse buttons	**Cui | Mouse Buttons**	**Customize | Mouse**
Plot styles	**PlotStyle**	**PlotStyle**
Quad Cursor	...[2]	Customize | Quad
Quick Access toolbar	Cui | Quick Access Toolbars	...[3]
Quick Properties palettes	Cui | Quick Properties	...[3]
Ribbon	Cui | Ribbon	Customize | Ribbon
Rollover tooltips	Cui |Rollover Tooltips	...[3]
Scripts	Script ', ActRecord	Script '
Selection previews	Options | Selection	Settings | Selection Preview
Shell commands	**...**[1]	**Customize | Shell Commands**
Shortcut/Context menus	**Cui | Shortcut Menus**	**Customize | Menus**
Status bar	Right-click, Diesel	Right-click, Diesel
System Variables	**SetVar, Options**	**SetVar, Settings**
Tablet	Cui | Legacy | Tablet	Customize | Tablet
Tool palettes	**ToolPalettes, Customize**	**ToolPalettes**
Toolbars	**Cui | Toolbars**	**Customize | Toolbars**
UCS icon	USCicon	Settings | User Coordinate System
User profiles	**Options | Profiles**	**ProfileManager**
Workspaces	Cui | Workspaces	Customize | Workspaces
3D Mouse	Through mouse driver	Through mouse driver

Notes:
[1] File must be edited outside of AutoCAD or BricsCAD with a text editor, such as Notepad
[2] Not available in AutoCAD
[3] Not available in BricsCAD

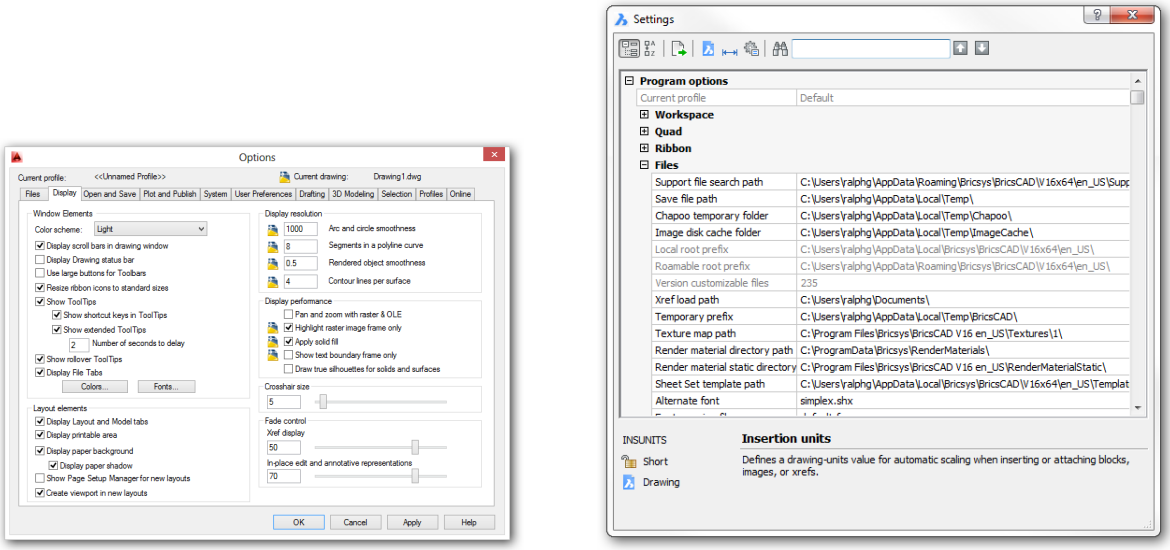
AutoCAD Options vs BricsCAD Settings

BricsCAD provides a set of extensive options for controlling your drafting environment, Just like AutoCAD — everything from modifying the look of the user interface to specifying names of project folders. Most settings are stored in system variables that have the same names as in AutoCAD, as well as in data files, many of which are compatible with AutoCAD.

TIP Chapter 5 provides information and tutorials on moving customization files from AutoCAD to BricsCAD.

SYSTEM VARIABLES AND PREFERENCES

AutoCAD's primary interface for changing settings is the dialog box displayed by the **Options** command. It provides access to many — but not all — system variables. In BricsCAD, the equivalent dialog box is called up by the **Settings** command. See Chapter 2 for more on these important dialog boxes.



Left: *AutoCAD's Display tab in the Options dialog box;* ***right:*** *BricsCAD's Files node in the Settings dialog box*

BricsCAD supports most of AutoCAD's system variables; in addition, it has an further set of variables that it calls "preferences." *Preferences* operate just like system variables. Bricsys gave them the different name to indicate they are unique to BricsCAD. (See Appendix B for the complete list of sysvars and preferences.)

Both CAD programs allow you to enter the names of sysvars and preferences directly at the command prompt. The old **SetVar** command is available also. In addition, BricsCAD exports all the names and settings to a CSV file through an option in the Settings dialog box. (AutoCAD does not do this; instead, use the Logfileon command to record the output from the SetVar * command.)

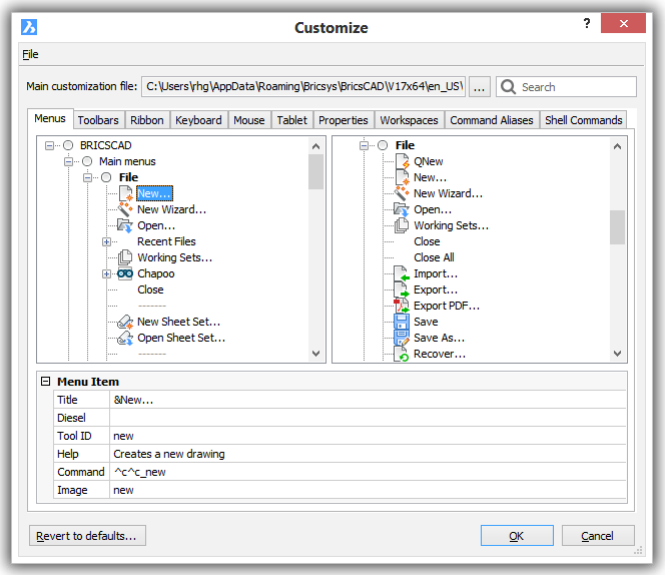
FILE PATHS

BricsCAD and AutoCAD drawings use many support files, such as fonts, profiles, and external references. Both CAD programs let you specify alternative paths to these folders, which means they can share each other's support files.

For more information on this capability, see "Common Operations through File Paths" in Chapter 5.

AutoCAD Cui vs BricsCAD Customize

The BricsCAD command **Customize** is equivalent to AutoCAD's Cui command. ("Cui" is available as an alias in BricsCAD.) The command displays a dialog box that centralizes customization of many BricsCAD user interface elements.



Customize handles many aspects of BricsCAD customization in a single dialog box

- **Menus** tab customizes the menu bar, menus, and context menus (shortcut menus)
- **Toolbars** tab customizes the toolbars and buttons
- **Ribbon** tab customizes tabs and panels
- **Keyboard** tab customizes the keyboard shortcuts
- **Mouse** tab customizes the mouse buttons, double-click actions
- **Tablet** tab customizes the tablet overlay menus and stylus buttons
- **Properties** tab customizes Quick Properties displayed by the Quad
- **Workspaces** tab customizes the quad cursor context
- **Aliases** tab customizes the command aliases
- **Shell Commands** tab customizes the shell commands

(NEW TO V17) The content of the Quad tab is moved to the Workspace tab.

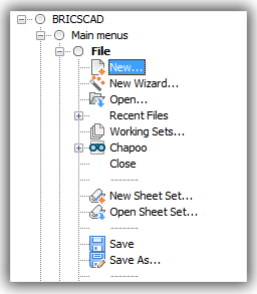
The process for customizing each of these elements is nearly always identical. This means that when you learn how to customize one element (such as a menu), you know how to do any other customization, such as a context menu or toolbar.

The way that BricsCAD customizes is, however, different from AutoCAD. So in this chapter I show you an example of customizing a BricsCAD menu.

UNDERSTANDING BRICSCAD'S CUSTOMIZATION TREE

To access the Customize dialog box, enter the **Customize** command, the **Cui** alias, or from the **Tools** menu choose **Customize**. Alternatively, right-click any toolbar or ribbon and then select **Customize**.

When you look at the Customize dialog box, one of the first things that stands out are the ● gray dots that prefix some menu items. Dots indicate *container* items, which are menu items that contain other items. For example, the File menu contains file-related items.



Large gray dots indicating container items

Here are more examples of containers:

- **BRICSCAD** container holds the names of the *menu groups*. These groups include things like "Main Menus" (the menu items seen on the menu bar) and "Context Menus," which appear when you right-click entities.
  - **Main Menus** is a container that holds items that appear on the menu bar, such as "File" and "Edit."
    - **File** is a container for the first menu appearing on the menu bar and holds items like "New" and "Open."
    - **Edit** is the container for the second menu on the menu bar.

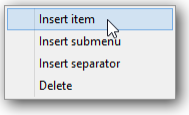
The row of five dashes "-----" indicates the position of a *separator bars* the gray lines that separate groups of menu items.

When you see the ⊞ and ⊟ buttons (*nodes*), these hold other containers or even submenus. Click a ⊞ node to expand the section; click the ⊟ node to close it again.

How BricsCAD Customizes Menus

The menu bar and its menus are customized in BricsCAD through the Customize dialog box's **Menu** tab. This is where you add, edit, and remove items to and from menus.

Most actions are performed through shortcut menus, like the one shown below. To do so, move the cursor into the Customize dialog box, right-click an existing menu item, and then choose an option from the shortcut menu that appears.



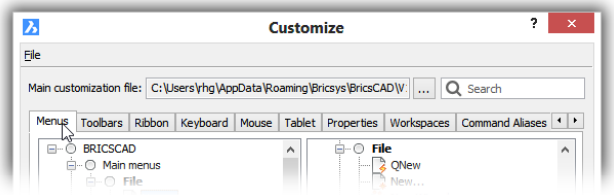
Right-clicking is how things get done in the Customize dialog box

Tutorial: How to Add a Command to a Menu

In this tutorial, you add the **CloseAll** command to the File menu. The CloseAll command closes all open drawings. It is to be located after the Close item.

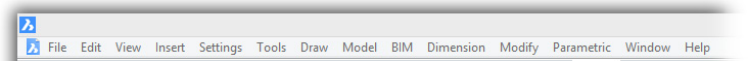
To add the command to the File menu, follow these steps:

1. To open the Customize dialog box, enter the **Customize** command.
2. When the dialog box appears, click the **Menus** tab.



Accessing the Menu tab in the Customize dialog box

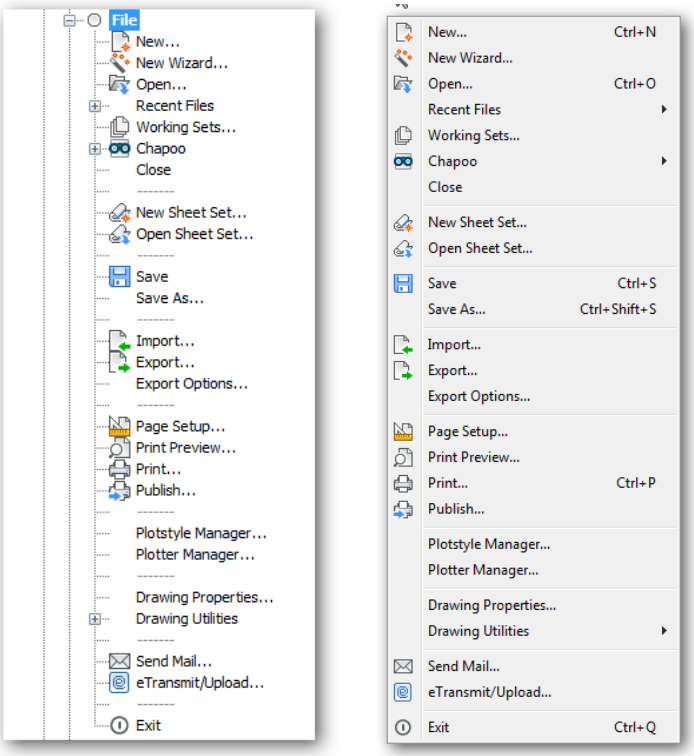
The **Main Menus** node defines the structure of the currently-loaded menu. Names like File, Edit, and View match the names on BricsCAD's menu bar. Some editions of BricsCAD may have names that are different from what is shown here.



Names on the menu bar matching the list in the Customize dialog box, one for one

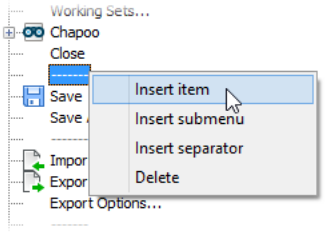
3. To open a container, click a ⊞ button. For this tutorial, you click the ⊞ next to the **File** container.

This reveals the items in the File dropdown menu, as illustrated at left below; the equivalent menu is shown on the right.



Left: *File menu container displayed by Customize dialog box;* ***right:*** *Menu items under the File dropdown menu*

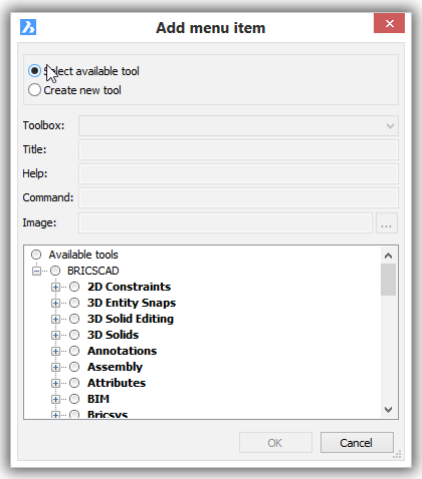
4. Move the cursor over the **-----** (separator) item located below **Close**. You choose this spot, because BricsCAD places new menu items *above* the current one.
5. Right-click (press the right mouse button). Notice that the shortcut menu commands for adding and removing menu items.



Inserting an item above the selected one

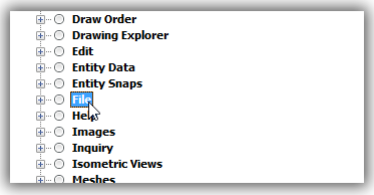
6. From the shortcut menu, choose **Insert Item**. This action adds a new menu item *above* the currently-selected one, the separator line ------.
7. Notice that BricsCAD opens the Add Menu Item dialog box, which lists all commands available in BricsCAD. From this list, you can select existing commands with **Select Available Tool** — or create macros with **Create New Tool**.

a. Choose the **Select Available Tool** option to access all of BricsCAD's built-in commands. (The other option, **Create New Tool**, is for creating macros — two or more commands strung together.)



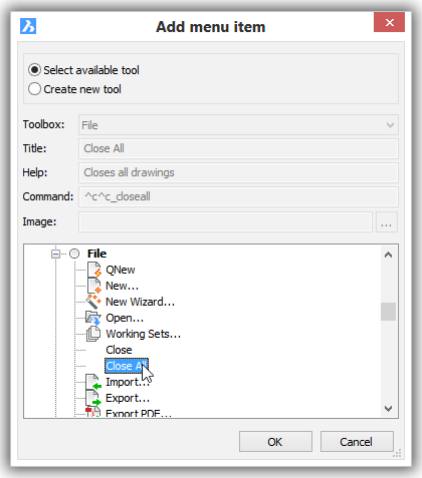
Dialog box for creating new menu items

b. Under Available Tools, scroll down to the **File** item. The fast way to get there is to click any item in the list (such as "2D Constraints"), and then tap the **F** key on the keyboard.



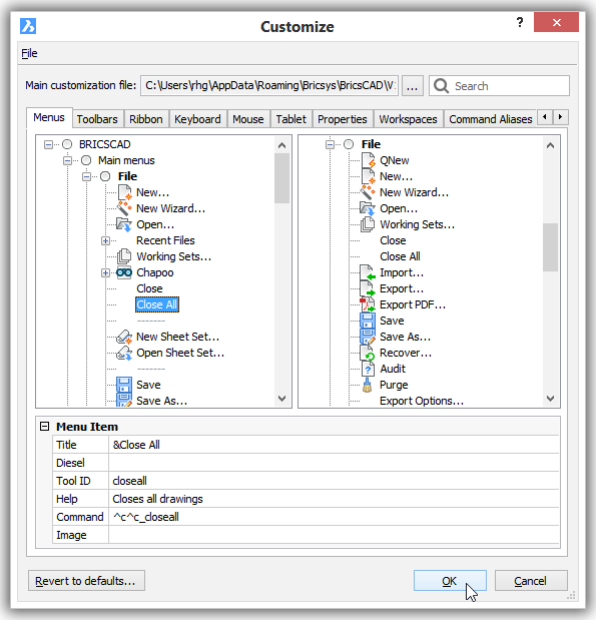
Getting to the File item

c. Under File, choose **Close All**. Notice that in the upper half of the dialog box BricsCAD fills in most of the parameters, such as Title, Help, and so on.



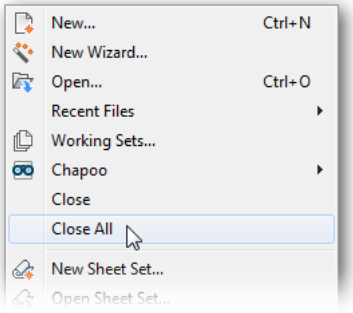
Choosing "Close All" from the list of a available commands

c. Click **OK**. Notice that the "Close All" command is added to the list under Close.



Close All command added to File menu

8. To ensure a new command actually works, always test your work, like this:
   a. Close the Customize dialog box by clicking **OK**.
   b. Choose the **File** menu. Notice that the "Close All" item has been added.



Testing the Close All command

   c. Click **Close All**. Does it work correctly? It should prompt you to save all open drawings that have changed since being loaded.

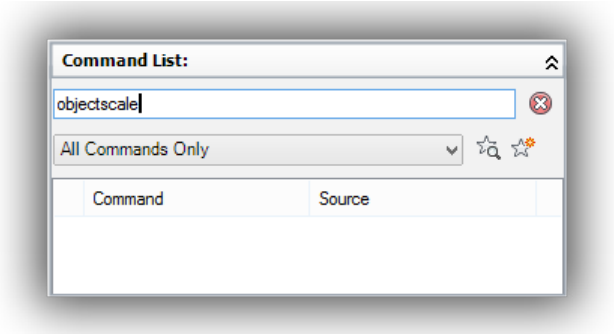
TIPS FOR WORKING WITH BRICSCAD MENUS

From my experience in customizing BricsCAD, here are answers to questions users might have.

Q: Which commands can be add to menus, toolbars, and so on?

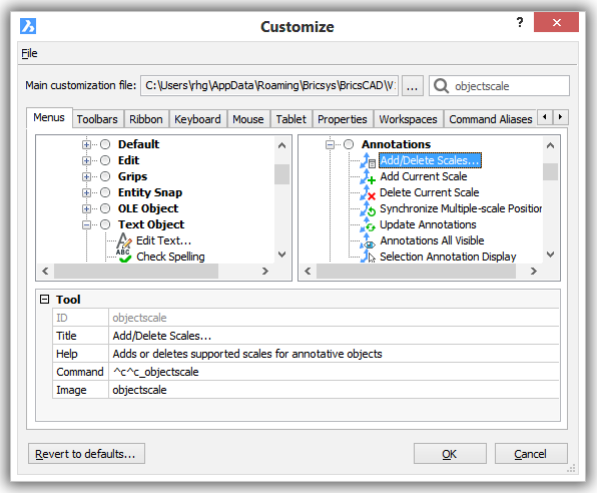
All commands. To see all commands in BricsCAD, peruse the list found the Customize dialog box.

To find a specific command, use the **Search** field. Happily, BricsCAD does not make the same error as AutoCAD, which searches only for *menu* names, not *command* names; you have to know that before you can search. For example, if you search AutoCAD's CUI for the "ObjectScale" command name, you won't find it; you have to search for "Add Object Scale," for that is the command's menu name.



AutoCAD unable to find commands by name in CUI

The good news in BricsCAD is that your search for the ObjectScale command name will be fruitful, for BricsCAD will find it.



BricsCAD finding commands by name

Q: What's the difference between "Insert" and "Append"?

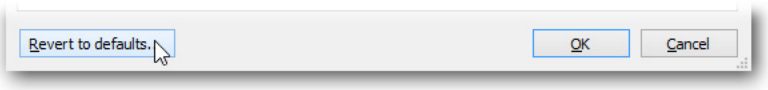
Sometimes a shortcut menu shows the verb **Append**, other times it shows **Insert** instead. The words seem similar, but have different actions; here's how:

- **Append Item** adds the new item at the *end* of the menu container
- **Insert Item** adds the new item *before* the currently-selected item

The difference does not matter much, because if an item ends up in the "wrong" location, you can just drag it to the correct position.

Q: What do I do when I mangle a customization?

Click the **Revert to Defaults** button found at the bottom of the Customize dialog box in BricsCAD. Be careful, though, because it removes *all* customizations you made to BricsCAD in this dialog box — except for the ones in the partial CUI files. So, always work with partial CUIs!



The nuclear option

CREATING A NEW MENU ITEM IN BRICSCAD

You add new "commands" through *macros*, which BricsCAD calls "tools." In this tutorial, you learn how to create a tool in BricsCAD that consists of two commands: the first saves the current drawing and then the second opens the Print dialog box. I've named the macro "Save'n Print," and it looks like this:

```
^C^C_qsave;_plot
```

Notice that the format of the macro is exactly the same as in AutoCAD.

You use BricsCAD's **Insert Tool** to add your own commands. These custom commands (a.k.a. "macros") are constructed from other commands, LISP routines, metacharacters, and Diesel instructions — just as in AutoCAD.

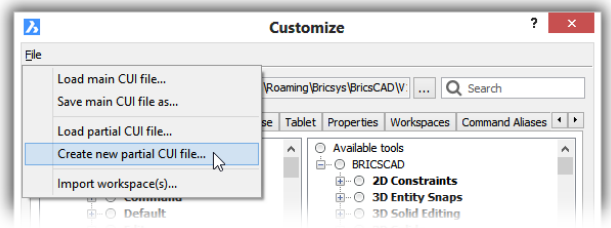
TIP The correct way is to add commands is to first create a new partial menu for them. The reason you do this is because of the **Revert to Defaults** button, which you see at the bottom of the Customize dialog box. Should a user (or you, even) click this button, then all customizations are lost! Except, of course, those added to partial menus.

The following tutorial shows you how construct macros for partial menus in BricsCAD.

Step 1: How to Create Partial Menus in BricsCAD

First, create the new partial menu, as follows:

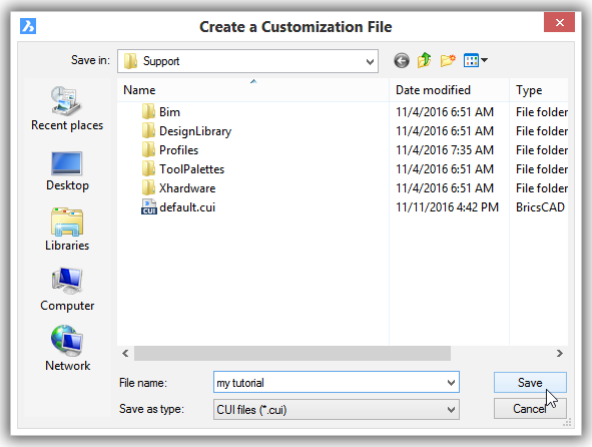
1. In the Customize dialog box, click **File**, and then choose **Create New Partial Cui File**.



Creating a new partial CUI file

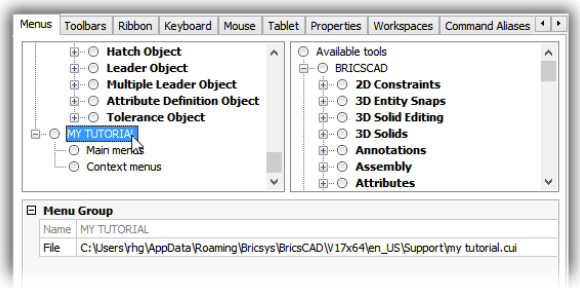
2. Notice the Create a Customization File dialog box. In the **File Name** field, enter a name that is brief but descriptive. For this tutorial, enter "my tutorial."

```
File Name        my tutorial
```



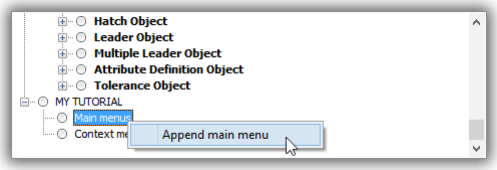
Naming the new partial menu file

3. Click **Save**. In the Customize dialog box, notice that "My Tutorial" is added as a node under the Files tab. The "My Tutorial" partial menu is also added to the Toolbar, Ribbon, and other tabs so that it can be used everywhere.



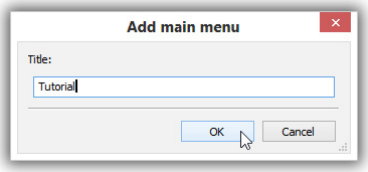
My Tutorial added to menus

4. Under My Tutorial, right-click **Main Menus**, and then choose "Append Main Menu."



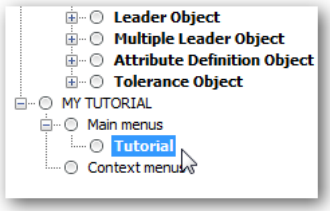
Adding a new main menu item

5. The Add Main Menu dialog box opens. Give the new menu its name, like "Tutorial," and then click **OK**.



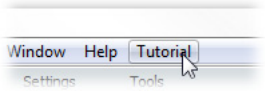
Naming the new menu item

After you clock OK to close the dialog box, the new Tutorial menu item appears.



Tutorial menu item added to the tree

6. To see this item on the menu bar, close the Customization dialog box by clicking **OK**. Notice that "Tutorial" appears after the Help item and is empty.



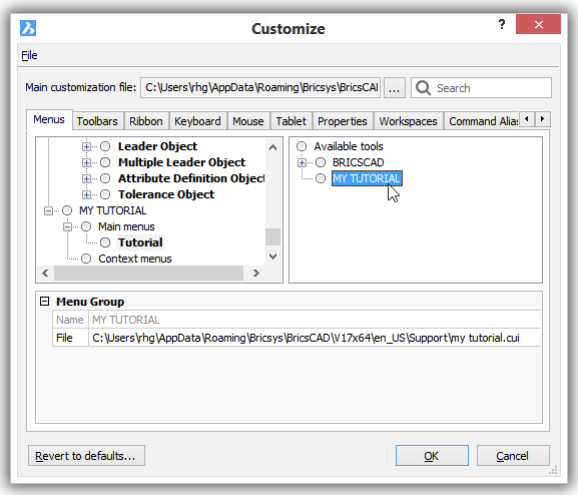
The menu bar showing the new Tutorial item

The partial menu is ready for the next step: adding custom commands.

Step 2: Adding Custom Commands to BricsCAD

To create a new custom command in BricsCAD, return to the Customize dialog box, and then follow these steps:

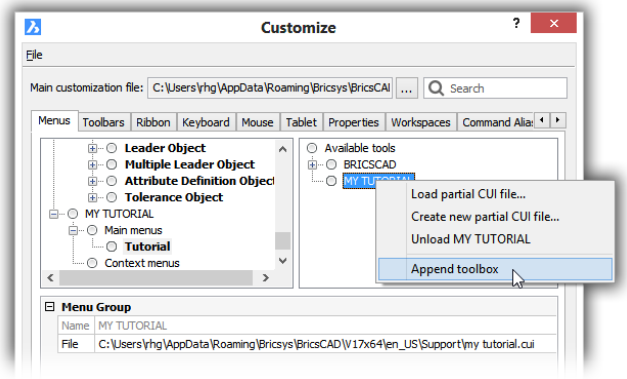
1. Look at the right-hand side of the Customize dialog box. In the **Available Tools** pane, navigate to the "My Tutorial" partial menu.



Working in the Available Tools pane

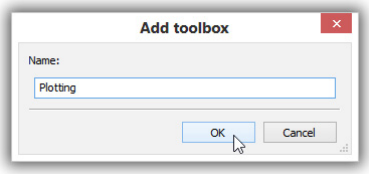
2. In partial menus, new commands are collected into "toolboxes." A *toolbox* is a collection of similar commands, such as ones related to editing or to file management.

   To add a toolbox, right-click "My Tutorial," and then from the shortcut menu, choose **Append Toolbox**.



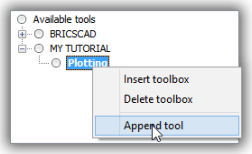
Adding a new toolbox to a new partial menu

3. In the Add Toolbox dialog box, enter "Plotting," and then click **OK**.



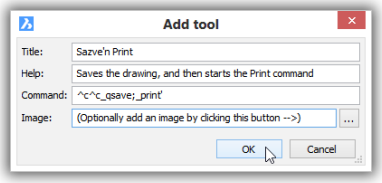
Naming the new toolbox

4. The toolbox created. Go ahead and create the new *tool*. (A tool is a command.) Right-click and choose **Append Tool**.



Adding a tool to the toolbox

5. The Add Tool dialog box appears. Here you define the new tool. Enter the macro in the **Command** field, as shown below.

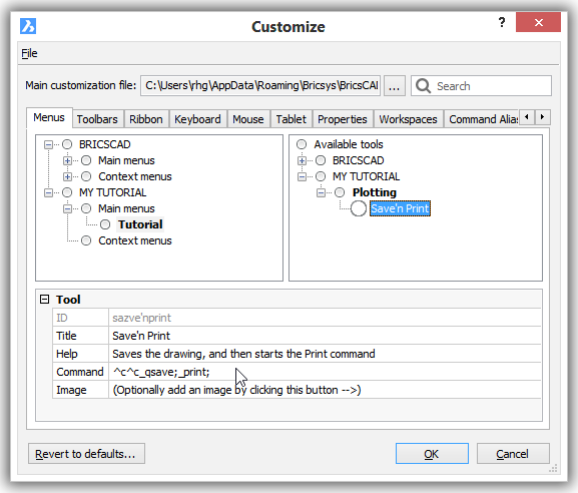


Fields for defining parameters of new tool

You can use the table below as a guide.

Parameter	Text that You Enter	Notes
Toolbox	File	Adds the new command to the File category of available tools
Title	Save'n Print	Specifies the name that appears in the File menu
Help	Saves the drawing, and then starts the Print command.	Specifies the help text that appears on the status bar
Command	^C^C_qsave;_print	Specifies the macro that cancels the current command, saves the drawing, and then starts the Print command
Image	*(leave blank)*	Specifies the icon, although none is required for menus

6. Click **OK** to exit the Add Menu Item dialog box. Notice that the new tool is added to the Tutorial menu (in the left pane of the Customize dialog box), as well as to the list of Available Tools (in the right pane).



New command appears in both panes

In addition, its parameters are shown in the Menu Item pane at the bottom of the dialog box. Here, you can edit the parameters, just as you can with regular commands.

7. Click **OK** to exit the Customize dialog box.
8. Test the new item by selecting **Save'n Print** from the **Tutorial** menu.

About BricsCAD's Macro Metacharacters

Menu items execute macros, which can contain *metacharacters*. BricsCAD and AutoCAD use many of the same metacharacters. I've listed some of the most common ones here so that you can see they are indeed identical:

Metacharacter	Meaning
^C	Cancels the current command.
'	Executes the command transparently.
_	Internationalizes the command.
;	Executes Enter.
\	Pauses the macro.

About BricsCAD's Menu Design Conventions

BricsCAD and AutoCAD use many of the same conventions for designing menus. Two of them are summarized below:

& (*ampersand*) designates shortcut keystrokes for accessing menu items with the **Alt** key

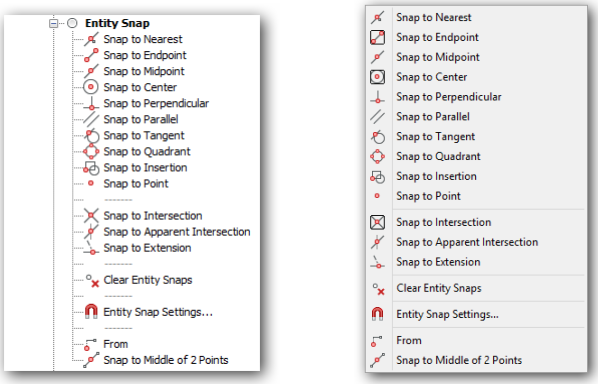
... (*ellipsis*) indicates the menu item will display a dialog box

About Diesel and DCL in BricsCAD

AutoCAD and BricsCAD employ the same Diesel expressions in menu macros and LISP routines, and the same DCL (dialog control language) code for constructing dialog boxes.

CUSTOMIZING CONTEXT MENUS

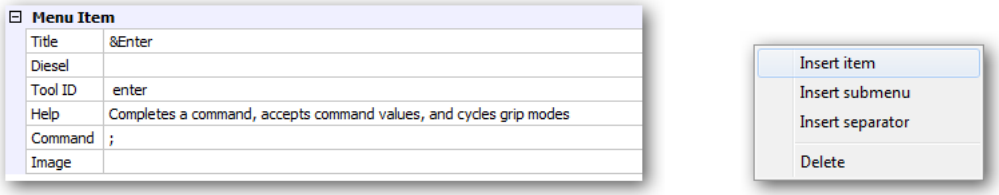
BricsCAD calls shortcut menus "context menus," because the menus change their content depending on the context. Context menus are found in the **Menus** tab, below the **Main Menus** section.



Left: *Defining the Entity Snap shortcut menu in the Customize dialog box of BricsCAD.;* ***right:*** *BricsCAD's Entity Snap context menu.*

As with menus, the list of items in each context container matches that of the shortcut menu. For instance, when you right-click, BricsCAD displays the Entity Snap context menu.

To customize a context menu, you have same options as you have with menus:



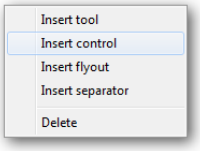
Left: *Pane for customizing a context menu item in BricsCAD;* ***right:*** *Shortcut menu for adding elements to context menus in BricsCAD.*

CUSTOMIZING TOOLBARS

Toolbars are customized in BricsCAD using the **Toolbar** tab of the Customize dialog box.

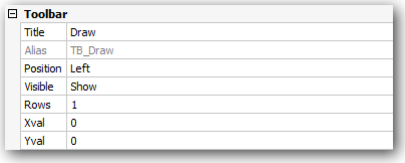
The process for customizing toolbars is identical to that of customizing menus, with two exceptions:

- Submenus of toolbars are called "fly outs."
- Toolbars can contain "controls," which menus cannot; control is another name for *droplist*.



Inserting controls or droplists into a toolbar

You can specify parameters for each toolbar and for each button. As in AutoCAD, BricsCAD can specify the initial location and visibility of toolbars. To do so, (a) select a toolbar name, such as Standard, and then (b) edit the settings in the pane, as shown below:

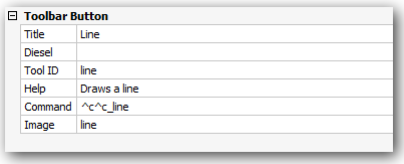


Parameters for positioning toolbars

The parameters for setting the initial position of toolbars are as follows:

Parameter	Options
Position	Floating Top Left Bottom Right
Visible	Show Hide

To edit individual buttons, select a name, and then edit the properties:



Parameters for toolbar buttons

Similar properties are available for flyouts.

TIP In AutoCAD and BricsCAD, the visibility of toolbars is controlled by the current workspace; all of AutoCAD's toolbars are turned off by default.

CUSTOMIZING RIBBON TABS AND PANELS

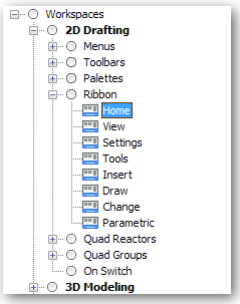
The ribbon's tab and panel elements are customized in BricsCAD with the **Ribbon** and Workspace tabs of the Customize dialog box.

The design of tabs and panels in BricsCAD is identical to those of Word, AutoCAD, and so on:

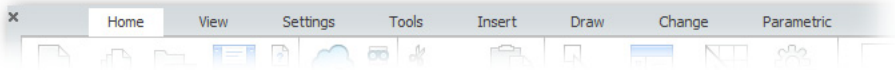
- The ribbon is segregated into one or more "tabs"
- *Tabs* contain one or more "panels"
- *Panels* contain one or more command elements, such as buttons and droplists

Customizing the ribbon takes place in three areas:

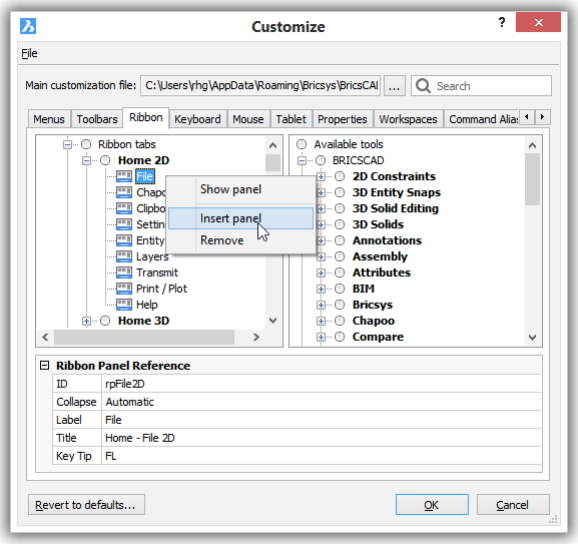
Ribbon. The look of the ribbon is specified by the Workspace tab, where all that happens is the names of tabs to be shown by the named workspace are listed. The figure below shows the names of tabs to be displayed the "2D Drafting" workspace.



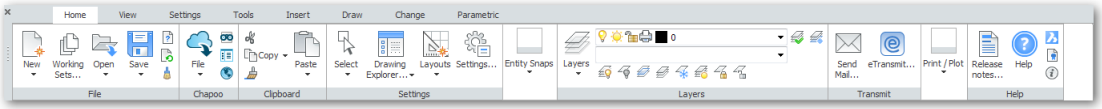
Above: *List of tabs to be displayed by the ribbon...*
Below: *...and the names of tabs on display in the ribbon*



Tabs. Just as a ribbon is just a list of tab names, a tab is just a list of panel names. These are customized by the **Ribbon Tabs** section of the Ribbon tab.

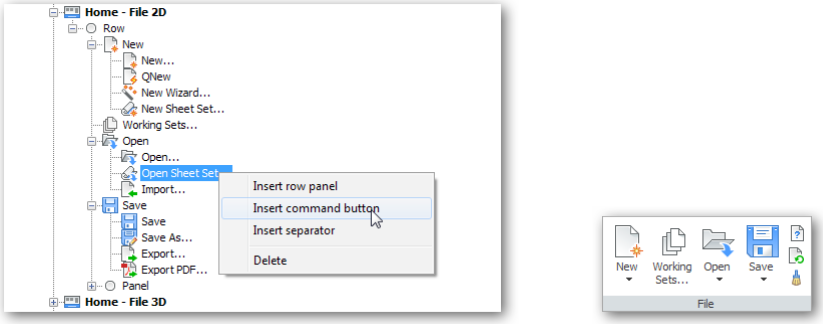


Above: *List of panels to be displayed by the Home 2D tab...*
Below: *...and the names of panels on display in the tab*



Use the right-click shortcut menu to insert and remove panels. To change the order in which panels appear in the tab, just drag them up and down the list.

Panels. The hard work takes place in designing the panels, as a ribbon can have big and little buttons with and without text labels, buttons strung horizontally or stacked vertically, droplists, and so on. They are customized by the **Ribbon Panels** section of the Ribbon tab:

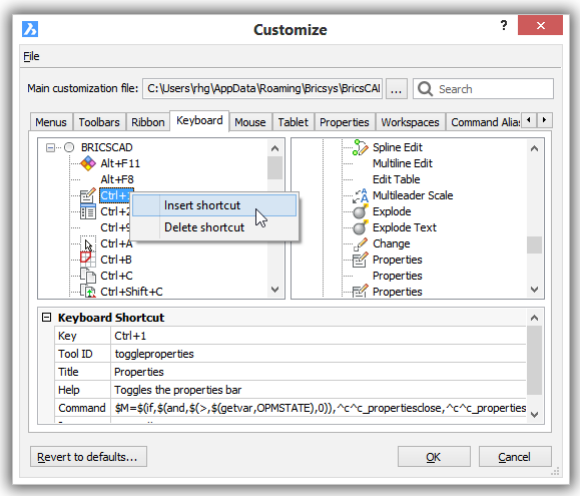


Left: *List of commands to be displayed by the File 2D panel...* ***right:*** *...and the buttons in the panel*

I won't go into the details here; they are best left to our *Customizing BricsCAD* book.

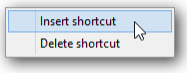
CUSTOMIZING KEYBOARD SHORTCUTS

Keyboard shortcuts are customized in BricsCAD by the **Keyboard** tab, as shown in the screen grab below. BricsCAD has many of the same shortcuts as does AutoCAD; see Appendix D for a useful cross-reference of all keystroke shortcuts used by both programs.



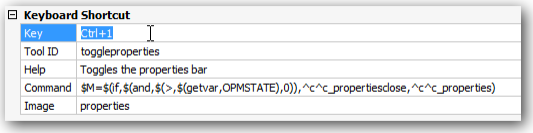
Customizing shortcut keystrokes

To add and remove shortcuts (or edit their assigned actions), right-click an existing one and then choose an option from the context menu:



Adding and removing keyboard shortcuts

Adding (inserting) shortcuts follows the same steps as adding menu items. You can enter the following kinds of shortcuts in the **Key** field, highlighted in the figure below:

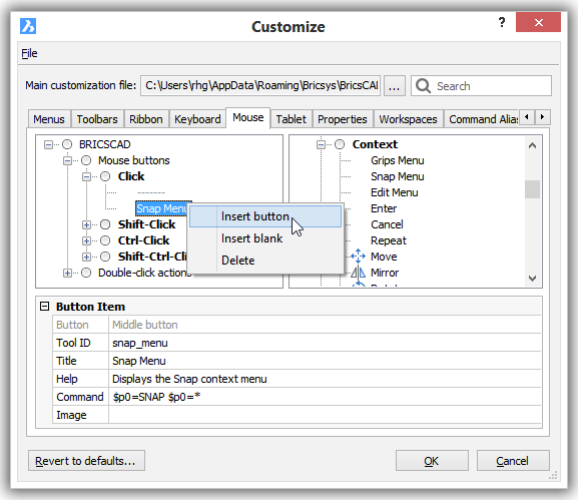


- **CTRL** keys
- **SHIFT**+**CTRL** keys
- Function keys
- **SHIFT**, **CTRL**, **ALT**, **CTRL**+**ALT**, **SHIFT**+**ALT**, **SHIFT**+**ALT**, and **SHIFT**+**ALT**+**CTRL** function keys

BricsCAD does not, unfortunately, warn you if a key combination is already in use.

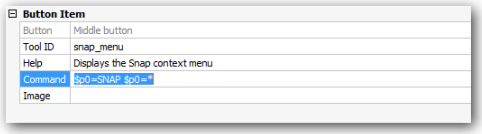
CUSTOMIZING MOUSE BUTTONS & DOUBLE-CLICKS

The actions of mouse buttons are customized in BricsCAD by the **Mouse** tab, as are double-click actions, as shown by the figure below:

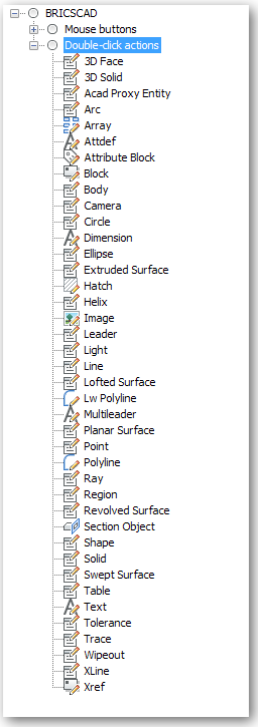


Customizing mouse button actions

To have the click of a mouse button display a menu to the user, you employ the same macro construction as in AutoCAD. See the code highlighted in the figure below:



Editing actions for mouse buttons



Double-Click Actions

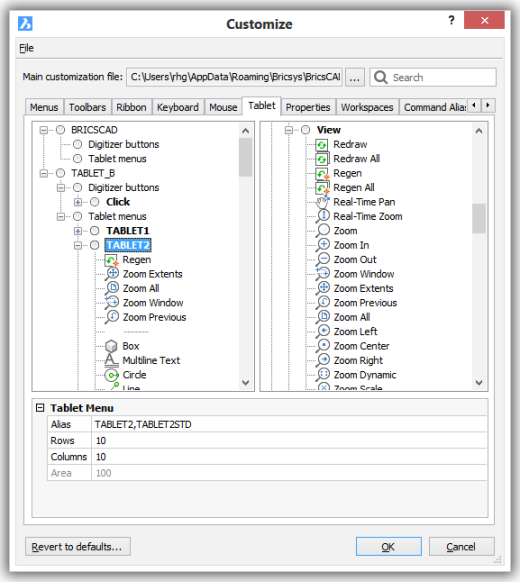
Double-click actions in BricsCAD are also customized with the **Mouse** tab. Double-click actions are customized in BricsCAD in the Mouse tab of the Customize dialog box. Go down to the **Double-click actions** section, and then edit the **Command** field.

In general, double-clicking an entity causes the Properties pane to appear, but this can be changed; indeed, Bricsys has assigned a number of other commands to the double-clicking of specific entities. Double-clicking a hatch pattern, for example, executes the HatchEdit command.

The lists of double-clickable entities is nearly identical for BricsCAD and AutoCAD. Now, some AutoCAD entities are not native to BricsCAD. As in AutoCAD, you can add and remove double-click actions to and from BricsCAD. To do so, right-click an existing action and then choose an option from the context menu. Inserting a double-click action takes the same steps as adding a menu item; see “Creating a New Menu Item” earlier in this chapter.

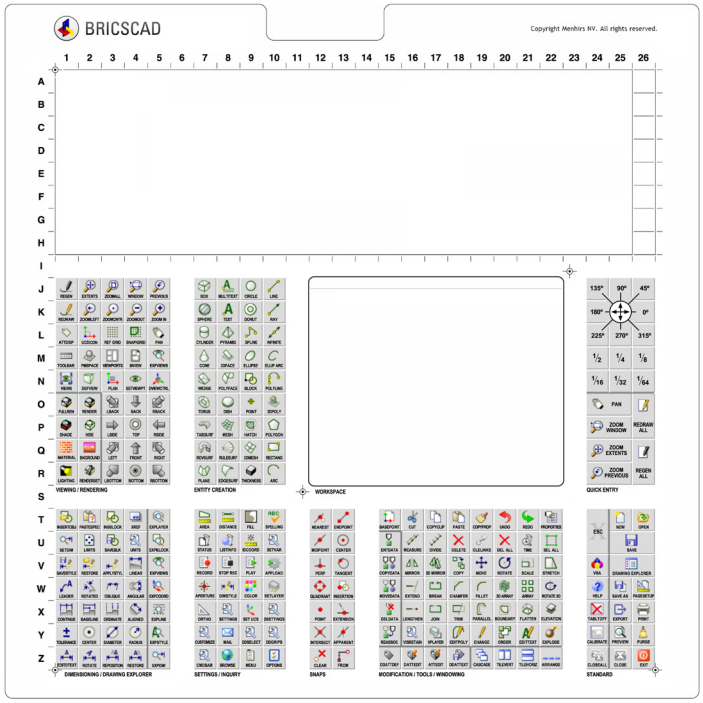
CUSTOMIZING TABLET BUTTONS AND MENUS

Tablet overlay menus and digitizer buttons are customized in BricsCAD through the **Tablet** tab, as illustrated below. Entries under Digitizer Buttons and Tablet Menus initially look empty because no tablet menu is loaded with the Default profile. To add tablet support to BricsCAD, download CUI files and drawings for tablet buttons and overlays from www.bricsys.com/bricscad/tools/Tablet.zip .



Tablet items appear after the partial CUI file for tablets is loaded in BricsCAD

The tablet overlay drawing provided by Bricsys is illustrated below:

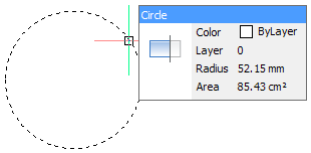


Tablet drawing provided by Bricsys containing the same commands as AutoCAD

After downloading, load the *tablet.cui* or *tablet(acadLike).cui* partial CUI files into BricsCAD with the MenuLoad command (just like you would in AutoCAD). Once one of these partial CUI files are loaded, then two sections in the Customize dialog box are filled with entries for tablet buttons and menus: Digitizer Buttons and Tablet Menus.

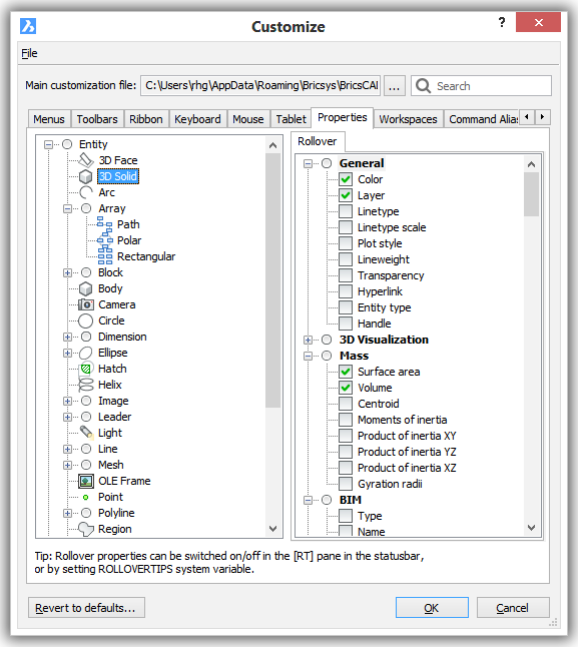
CUSTOMIZING QUICK PROPERTIES

Quick Properties in AutoCAD are displayed by a tooltip when the cursor hovers over an entity. In BricsCAD, they are displayed by the Quad.



Property information displayed by the Quad in BricsCAD

Just as in AutoCAD, you can specify (customize) the properties displayed for each and every entity type in BricsCAD. This is done through the **Properties** tab of the Customize dialog box.



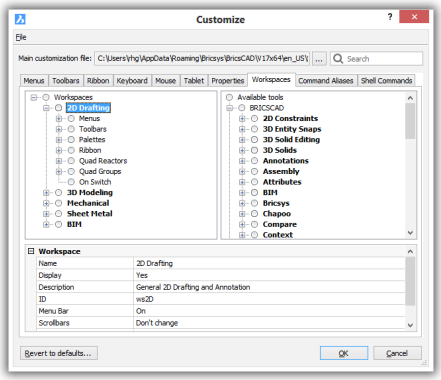
Selecting the properties to display for the 3D Solid entity

Select an entity in the left pane, and then choose which properties you want the Quad to display form the list in the right pane.

TIP If quick properties do not display in the Quad, click the **RT** button on the BricsCAD status bar.

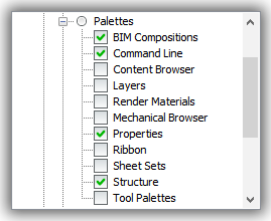
CUSTOMIZING WORKSPACES AND QUAD

Workspaces in AutoCAD and BricsCAD have the same effect on the user interface: they decide which toolbars, palettes, menu items, and ribbon tabs appear when users switch to a different workspace. Workspaces are customized in the **Workspace** tab of the Customize dialog box.



Customizing workspaces in BricsCAD

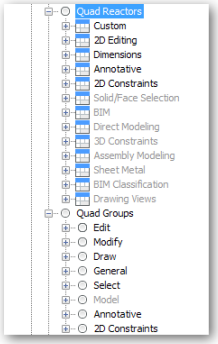
For instance, to decide which palettes (panels) should be displayed in the "2D Drafting" workspace, open the **Palettes** node, and then turn panels names on or off. It's that simple



Deciding which panels (palettes) to display

Quad

BricsCAD uses workspaces to also determine the look and functions of its unique Quad cursor. It is customized by the **Quad Reactor** and **Quad Groups** sections.

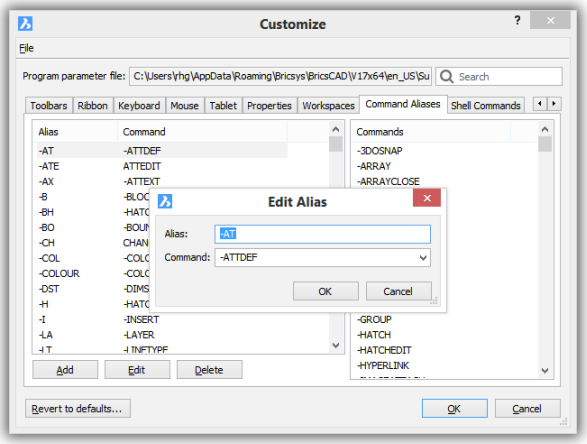


Customizing the Quad

CUSTOMIZING ALIASES AND SHELL COMMANDS

Command aliases are customized in BricsCAD with the **Aliases** tab. BricsCAD has many of the same aliases as does AutoCAD; see Appendix C of this ebook for the cross-reference.

In BricsCAD, you create and edit aliases inside the Customize dialog box. To do so, click the **Add** or **Edit** button to see the Edit Alias dialog box, illustrated above. For defining aliases and shell commands, BricsCAD uses the same format for aliases as does AutoCAD.



Editing an alias

Both CAD packages store the definition in a *.pgp* file but with different filenames:

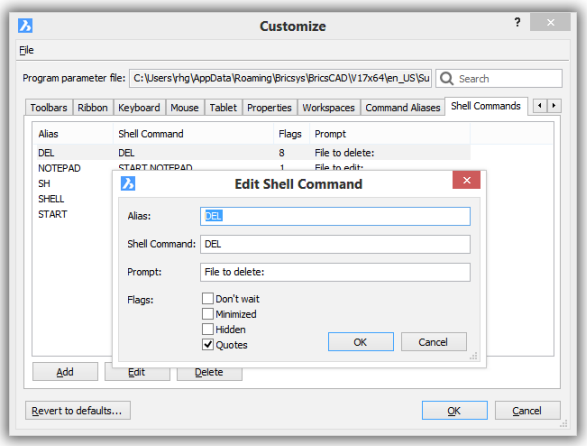
BricsCAD aliases are stored in the *default.pgp* file.

AutoCAD aliases are in the *acad.pgp* file

When you copy an *acad.pgp* to a BricsCAD installation, rename the incoming file "default.pgp."

Shell Commands

BricsCAD users the same format for shell commands as AutoCAD. They are customized in the **Shell Commands** tab, as illustrated below:

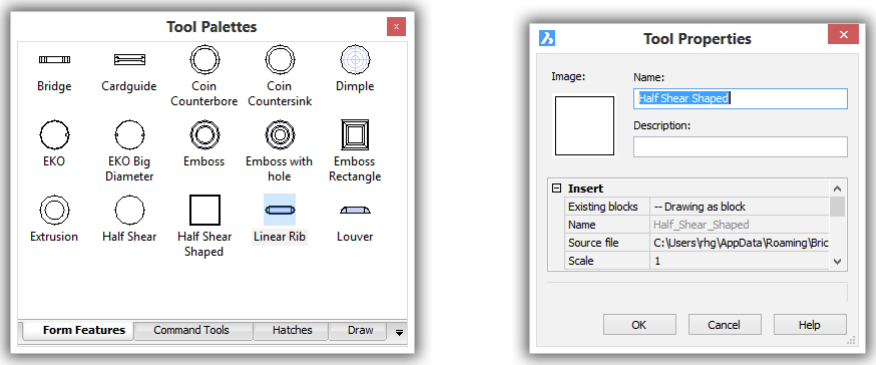


Editing a shell command

CUSTOMIZING TOOLS PALETTES

Tool palettes can be customized, but the process in BricsCAD is different than in AutoCAD. Here is the difference between the two CAD systems in how items are added to palettes:

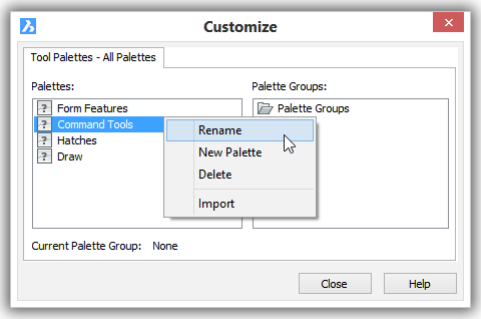
BricsCAD — you drag *commands* from the Customize dialog box to the Tools palette, as described below
AutoCAD — you drag *entities* from the drawing into the Tools palette



***Left:** Palette from BricsCAD; **right:** customizing the actions of an icon*

When it comes to palette groups, both CAD programs use a separate dialog box to create and change them, as well as to export and import palette definition files. Despite the presence of the dialog box, BricsCAD cannot, however, create groups, nor does it export palettes. To access the palette group dialog box:

BricsCAD — right-click the Tools palette, and then choose **Customize Palettes**
AutoCAD — enter the **Customize** command, which is unrelated to CUI



(NEW TO V17) *Customizing groups of palettes*

(Notice that BricsCAD has *two* dialog boxes named "Customize," one for customizing the UI and the other for palette groups!) Both programs store Tools palette definitions in external files in XML format:

- AutoCAD stores palette definitions in ATP files, short for "AutoCAD tool palettes"
- BricsCAD stores them in BTP files, short for "BricsCAD tool palettes." Both are XML-format files.

TIP Both CAD systems import palette definitions using XTP files, short for "Xml Tool Palette," which allows BricsCAD to read palettes from AutoCAD

Other Areas of Customization

Customizing BricsCAD doesn't just occury in the Settings and Customize dialog boxes. Here is a review of additional elements that can be modified, including fonts, linetypes, hatch patterns, and plot styles.

FONTS

AutoCAD and BricsCAD use the same types of font files:

- TrueType (*.ttf*) fonts
- Compiled shape (*.shx*) fonts

This means BricsCAD can use all of the fonts displayed by any AutoCAD drawing.

TrueType Fonts

All TrueType TTF files are stored in a common folder accessed by all programs. AutoCAD and BricsCAD both access the same source, and so there is no need to copy *.ttf* files to some BricsCAD folder.

- Windows stores TTF fonts in folder */windows/fonts*
- Linux stores TTF fonts in folder */usr/share/fonts/truetype*
- Mac stores TTF fonts in folder */System/Library/Fonts*

LEGALITIES: ABOUT COPYING FILES

Autodesk permits the copying of support files, since the corporation understands that drawings are effectively disabled when DWG files are sent to clients without these crucial files. Support files that are coded in ASCII contain the following notice from Autodesk:

> *Permission to use, copy, modify, and distribute this software for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and the limited warranty and restricted rights notice below appear in all supporting documentation.*

When copying files, do so in whole so that you include the notices that Autodesk asks you to preserve.

TRUETYPE FONTS

There is one exception. Some TrueType fonts (*.ttf* files) are commercial products, and cannot be copied without payment to the copyright holder. The good news, however, is that all TrueType fonts provided with Windows and AutoCAD may be copied freely. If a drawing contains copyrighted TrueType fonts, you can often find ones that look similar but cost nothing.

SHX Fonts

AutoCAD keeps SHX fonts in the *C:\program files\autodesk\autocad \fonts* folder. To use them with BricsCAD, you can copy the SHX files to the equivalent folder in Bricsys:

- Windows stores SHX fonts in folder *C:\Program Files (x86)\Bricsys\BricsCAD V17\Fonts*
- Linux stores SHX fonts in folder */opt/bricsys/bricscad/fonts*
- Mac stores SHX fonts in folder */Applications/BricsCAD V17.app/Contents/MacOS/Fonts*

AutoCAD also installs TTF versions of its SHX fonts in *\windows\fonts* folder, because TrueType fonts look much smoother and fill better than SHX fonts. If possible, you should use TrueType fonts in your drawings, instead of SHX fonts. While Autodesk continues to provide SHX font files, it only does so to provide compatibility with old drawings.

If necessary, use the *default.fmp* file to map SHX font names to TTF ones. See below.

PFB Fonts

AutoCAD also supports the rarely-used PostScript *.pfb* font format. The support is indirect: you have to use its Compile command to convert PostScript fonts into SHX format. BricsCAD does not work with PostScript fonts, but this does not matter as PFB files are actually as SHX fonts in AutoCAD drawing files.

PostScript fonts are the default for Linux, but this does not matter, because neither CAD package uses them directly.

Font Mapping

BricsCAD and AutoCAD support font mapping, something that becomes handy when a font is not displayed in a drawing. This occur when DWG files are copied from one computer to another, but the second computer doesn't have all of the font files needed by the drawings.

Here are two ways to use font mapping:

- *Quick'n dirty* method uses the **FontAlt** system variable to specify the name of a single font to use when the correct one(s) cannot be found. Only one font is substituted for all missing fonts.
  AutoCAD specifies *arial.ttf*, while BricsCAD uses *simplex.shx*.

- *Comprehensive* method uses the **FontMap** system variable to specify the name of a *.fmp* file, which holds a list of all font names that can be mapped to alternative. Here is where the file is located:

CAD System	FontMap	Default Folder
AutoCAD	*acad.fmp*	C:\Users\login\AppData\Roaming\Autodesk\AutoCAD \R20.0\enu\Support
BricsCAD Windows	*default.fmp*	C:\Users\login\AppData\Roaming\Bricsys\BricsCAD\V17\en_US\Support
BricsCAD Mac	*default.fmp*	Users\login\Library\Preferences\Bricsys\BricsCAD\V17x64\en_US\Suppot
BricsCAD Linux	*default.fmp*	home/login/Bricsys/BricsCAD/BricsCAD/V17/en_US/Support

Both CAD systems use the same simple format for FMP files: replacement font names are separated by a semi-colon, one per line. Here are the first few entries of the BricsCAD version of the file:

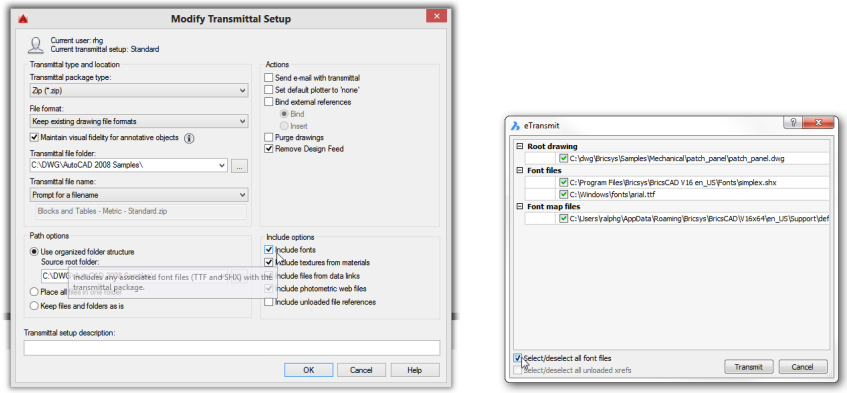
```
ic-comp;complex.shx
ic-complex;complex.shx
ic-gdt;gdt.shx
ic-ital;italic.shx
ic-italc;italicc.shx
```

Should you need to, copy the *acad.fmp* file from AutoCAD, rename it to *default.fmp*, and then paste it into the folder used by Bricsys.

eTransmit

One way to ensure that Bricsys has all the fonts it needs is to use AutoCAD's **eTransmit** command. This command collects the DWG file, needed support files, all font files, and any attachments, and then places them into a folder or a ZIP file.

There is just one problem: by default, the option to include font files is turned off — for legal reasons. (See the boxed text, "Legalities: About Copying Files.") To include fonts in AutoCAD, click the **Transmittal Setups** button, choose **Modify**, and then turn on the **Include Fonts** option. See figure below.



Left: *Including all fonts files in AutoCAD;* ***right:*** *including the font files in BricsCAD*

In BricsCAD, have eTransmit list all font files by turning on the **Select/Deselect All Font Files** option.

If you want just a list of needed fonts and other support files, click AutoCAD's **View Report** button, and you get a list of required and missing files:

```
AutoCAD Drawing Standards File References:
  MKMStd.dws

AutoCAD Font Map References:
  acad.fmp

AutoCAD Compiled Shape References:
  Fonts\txt.shx
  Fonts\romand.shx

The following files could not be located:
  @Arial Unicode MS.(shx,ttf)
  Textures\Mats\
  PlotCfgs\Sample Floor Plan_Base.stb
```

LINETYPES AND HATCH PATTERNS

BricsCAD and AutoCAD use the same definitions for linetypes, as well as for hatch patterns:

- **Simple** linetypes defined by *.lin* files
- **Complex** linetypes defined by *.lin* and *.shx* files
- **Hatch patterns** defined by *.pat* files

This means that BricsCAD can use linetypes and hatch patterns that have been customized for AutoCAD. AutoCAD stores LIN and PAT files in folders Windows stores the files in folder *C:\Users\<login>\AppData\Roaming\Autodesk\AutoCAD \R20.0\enu\Support.* BricsCAD stores LIN and PAT files in the following folders:

- Windows stores the files in folder *C:\Users\<login>\AppData\Roaming\Bricsys\BricsCAD\V17\en_US\Support*
- Linux stores the files in folder *home/<login>/Bricsys/BricsCAD/BricsCAD/V17/en_US/Support*
- Mac stores the files in folder */Users/<login>/Library/Preferences/Bricsys/BricsCAD/V17x64/en_US/Support*

Tutorial: How to Copy AutoCAD .**lin** and .**pat** Files to BricsCAD

If you wish to reuse linetypes and hatch patterns from AutoCAD, then follow these steps to copy and rename them:

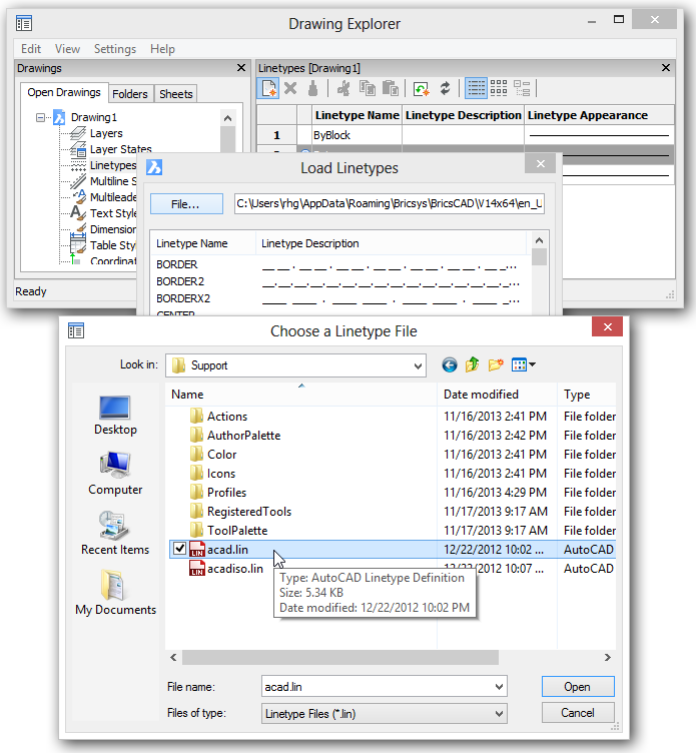
1. Copy the *.lin*, *.shx*, and *.pat* files from their AutoCAD support folder (see above for its location)...
2. ...to the BricsCAD support folder (see lists above for locations).
3. Once copied, however, you must to rename the files, because BricsCAD uses different file names for default linetype and hatch pattern files. For example, the *acad.lin* linetype file needs to be renamed to *default.lin*. Here is the entire list of file names:

File Type	AutoCAD Default Name	BricsCAD Default Name	Notes
Linetype definitions	*acad.lin*	*default.lin*	AutoCAD standard linetypes
	acadiso.lin	*iso.lin*	ISO-standard linetypes
	ltypeshp.shx	*ltypeshp.shx*	Shape files for complex linetypes
Hatch pattern definitions	*acad.pat*	*default.pat*	AutoCAD standard patterns
	acadiso.pat	*iso.pat*	ISO-standard hatch patterns

As an alternative to copying and renaming files, you could instead import AutoCAD linetype files into BricsCAD. The drawback is that this method works only on a per-drawing basis, yet could be useful for populating DWT template files. It works like this:

1. In BricsCAD, enter the **Linetype** command to open the Drawing Explorer window at the Linetypes node.
2. Click the **New** button to display the Load Linetypes dialog box.
3. Click **File** to access other *.lin* files.
4. Use the **Look In** droplist to navigate to AutoCAD's support folder, such as *C:\Users\<login>\AppData\Roaming\Autodesk\AutoCAD \R20.0\enu\Support*. Remember to replace *<login>* with your Windows login name.

5. Choose the *.lin* file you wish to open, and then click **Open.** The linetypes from AutoCAD are added to the current drawing.



Loading AutoCAD linetype files into the current BrisCAD drawing

Linetypes and hatch patterns are customized by BricsCAD and AutoCAD the same way, editing the related *.lin* and *.pat* files with Notepad or another text editor.

TIP To see custom hatch pattern files in BricsCAD, when their names differ from *default.pat*, set the hatch **Type** to "Custom" in the Hatch Pattern Palette dialog box.

PLOT STYLES

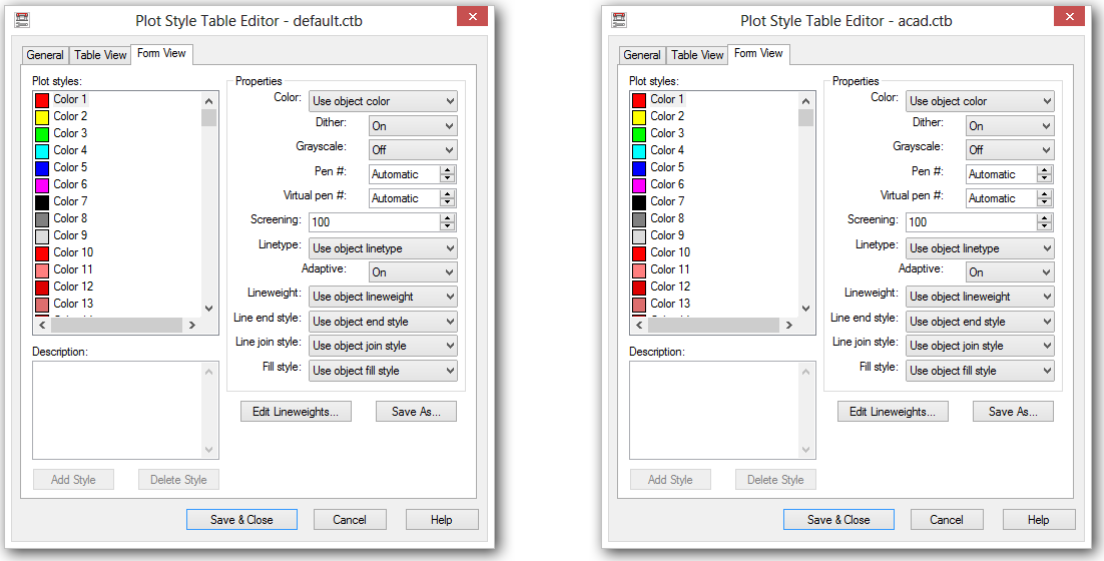
BricsCAD and AutoCAD support both color-based and style-based plot styles that allow entities to look different when plotted. Recall that CTB files are for the older color-based plot style tables, while STB files are for the newer style-based plot style tables. The figures below show that the style-based plot style tables of both CAD programs are identical:

This means BricsCAD can use STB and CTB files created by AutoCAD — after you rename them, because the sole difference is the file name of the default files:

BricsCAD default plot style file is *default.stb*

AutoCAD default plot style name is *acad.stb*

To create or edit plot styles in BricsCAD, use the **PlotStyle** command. Or choose **Plotstyle Manager** from the **File** menu.



Left: *BricsCAD's plot style table;* ***right:*** *AutoCAD's plot style table*

Plotter Manager

BricsCAD and AutoCAD both support PC3 plotter manager files, which allow us to customize plotter options. This means that BricsCAD can use PC3 files created in AutoCAD.

The plotter configuration editors of both CAD programs are similar. To create and edit plotters in BricsCAD, choose **Plotter Manager** from the **File** menu, or enter the **PlotterManager** command.

Supported Files

In addition to DWG drawing files, BricsCAD and AutoCAD employ many additional files. The following tables cross-reference by extension supported files between the two CAD packages.

Drawing Files

.adt	.adt	Audit log files
.bak	.bak	Backup drawing files
.dwf	.dwf	Design Web format files
.dwfx	...	XPS compatible version of DWF files
.dwg	.dwg	Drawing files
.dws	...	CAD standards files
.dwt	.dwt	Drawing template files
.dxb	...	Binary drawing interchange files for CAD/camera
.dxf	.dxf	Drawing interchange files, ASCII and binary
.sv$	.sv$	Autosaved drawing files

.xlg	.xlg	Xref log files
.$$$	...	Emergency backup files
.$ac	...	Temporary files created by AutoCAD
.$a	...	Temporary files

Support Files

.acb	...	AutoCAD color book files
.acl	...	Autocorrect list files
.arg	.arg	User profile files
.atc	.btc	AutoCAD / BricsCAD tool catalog files
.aws	...	AutoCAD workspace files
.blk	...	Block template files
.cfg	.cfg	Configuration files
.chm	chm	Compiled HTML format help files
.chx	...	Standards check files
.cui	.cui	Customize User Interface files
.cuix	...	Customization container files
.cus	.cus	Custom dictionary files
.dbq	...	Database query files
.dbt	...	Database template files
.dbx	...	Database extension files
.dct	.dic	Dictionary files
.dsd	...	Drawing set description files
.dst	.dst	Sheet set data files
.err	...	Error log files
.fdc	...	Field catalog files
.fmp	.fmp	Font mapping files
.hdi	...	Heidi device interface files
...	.hlp	Windows-format help files
.htm, .html	.htm, .html	Hypertext markup language files
...	.icm	IntelliCAD menu files
.ies	...	Illumination distribution data files
.ini	...	Configuration (initialization) files
.lin	.lin	Linetype definition files
.log	.log	Log files created by the LogFileOn command
...	.lwi	Base material files
.mli	...	Material library files for rendering
.mln	.mln	Multiline style files
.mnc	...	Compiled menu files (deprecated as of AutoCAD 2006)
.mnd	...	Uncompiled menu files containing macros (deprecated)
.mnl	...	AutoLISP routines used by AutoCAD menus (deprecated)
.mnr	...	Menu resource files
.mns	.mns	AutoCAD-generated menu source files (deprecated)
.mnu	.mnu	Menu source files (deprecated as of AutoCAD 2006)
.nfl	...	Filter list files
.pat	.pat	Hatch pattern definition files
.ptw	...	Publish to Web settings files
.pwt	...	Publish to Web template files

.rml	...	Redline markup files (obsolete)
.shp	...	Shape and font definition files
.shx	.shx	Compiled shape and AutoCAD font files
.slg	...	Status log files
.ttf	.ttf	Microsoft font files
.txt	.txt	Text message files
.udl	...	Microsoft data link files
.xml	...	Extended markup language files
.xmx	...	External message files
.xpg	...	XML-format tool palette group files
.xtp	.xtp	Tool palette exchange files

Plotting Support Files

.ctb	.ctb	Color-table based plot parameter files
.pc2	...	Plot configuration parameters files for AutoCAD 2000 (deprecated)
.pc3	.pc3	Plot configuration parameters files since AutoCAD 2000i
.pcp	...	Plot configuration parameters files for AutoCAD R14 (deprecated)
.plt	.plt	Plot files
.pmp	.pmp	Plotter model configuration files
.pss	...	Plot stamp settings files
.stb	.stb	Style-table based plot parameter files

Import-Export Files

.3ds	...	3D Studio files
.bmp	.bmp	Windows raster files (device-independent bitmap)
.cdf	.cdf	Comma delimited files
.dgn	...	MicroStation V8 and V7 design files
.dxe	...	Data extraction files created by DataExtraction command
.dxx	...	DXF files created by AttExt command
...	.ecw	Enhanced Compression Wavelet files
...	.emf	Enhanced meta format files
.eps	...	Encapsulated PostScript files
.fax	...	Fax raster plot files
.fit	...	FIT raster plot files
.gif	.gif	CompuServe image files
.jpg, .jpeg	.jpg, .jpeg	Joint photographic expert group files
...	.jp2	JPEG 2000 files
.kml	...	Google Earth files (keyhole markup language)
.kmx	...	Compressed KML files
.pcx	.pcx	Raster format files
.pdf	.pdf	Portable document format files
.png	.png	Portable Network Graphics raster files
.sat	.sat	ACIS solid object files (short for “Save As Text”) files
.sdf	.sdf	Space-delimited files
.slb	.slb	Slide library files
.sld	.sld	Slide files
.stl	...	Solid object stereo-lithography files
...	.svg	Scalable vector graphics

.tga	.tga	Raster format (Targa) files
.tif	.tif	Raster format (Tagged image file format) files
.txt	.txt	Space delimited files
.wmf	.wmf	Windows metaformat files
.xls	...	Excel spreadsheet files
API and Programming Files		
.actm	...	Active macro source code files
.arx	.tx	AutoCAD / Teiga runtime extension files
...	.brx	Bricsys runtime extension files
.cpp	.cpp	ObjectARX source code files
.dce	.dce	Dialog error log files
.dcl	.dcl	Dialog control language descriptions of dialog boxes
...	.drx	Design runtime extension files
.dll	.dll	Dynamic link libraries
.dvb	.dvb	Visual Basic for Applications program files
.fas	...	AutoLISP fast load programs files
.h	.h	ADS/SDS and ARX/BRX/TX function definition files
.lib	.lib	ARX BRX/TX function library files
.lsp	.lsp	AutoLISP/LISP program files
...	.mcr	Macro files
.pgp	.pgp	Program parameters files (external commands and aliases)
.rx	...	Lists of ARX applications that load automatically
.scr	.scr	Script files
.unt	.unt	Unit definition files
...	.vbi	VBA project files prior to BricsCAD V8
.vlx	...	Compiled Visual LISP files

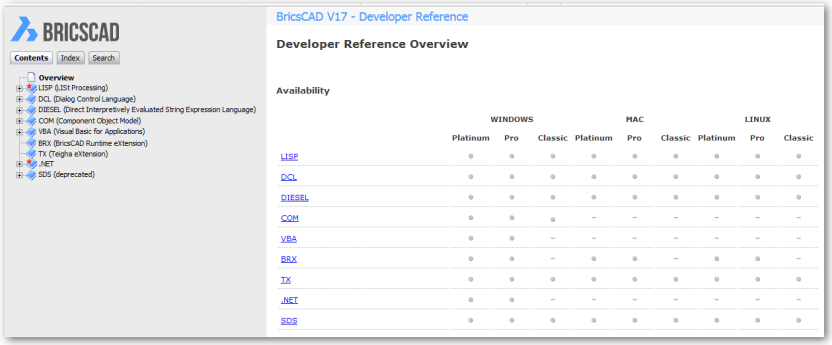
Programming Considerations

By supporting almost the same list of programming languages and APIs as does AutoCAD, Bricsys makes it easy for you to transfer your AutoCAD add-ons to BricsCAD:

AutoCAD API	Equivalent in BricsCAD	Notes
Action Recorder (*)	Scripts, SCR	AutoCAD's Action Recorder scripts cannot be edited; scripts recorded by BricsCAD can be edited.
ActiveX	ActiveX	In-place editing; not available in BricsCAD for Linux or Mac
ADS	SDS	ADS code ported from AutoCAD requires just a recompile using BRX headers; ADS/SDS are deprecated by Autodesk and Bricsys.
ARX	BRX or TX	Ported ARX code requires just a recompile using new BRX headers; when used with TX (ex-DRX), ported ARX code must be rewritten.
AutoLISP	LISP	Ported AutoLISP code runs as-is in BricsCAD; no changes needed, includes support for Vl, Vlr, Vla, and Vlax functions and encryption.
COM	COM	Ported AutoCAD COM code runs as-is in BricsCAD; not available in BricsCAD for Linux or Mac.
CUI	CUI	Ported AutoCAD CUI files made need adjsting for BricsCAD.
Diesel	Diesel	Ported Diesel code runs as-is in BricsCAD; no changes needed.
DCL	DCL	Ported DCL code runs as-is in BricsCAD; no changes needed.
CUI	CUI	Ported AutoCAD menu and toolbar macros work as-in in BricsCAD.
.Net	Teigha.NET	BricsCAD provides Teigha.NET and extra BRX-managed wrappers; not available in BricsCAD for Linux, Mac, or Windows Standard version.
...	TX	Teigha eXtensions (formerly DRX) from Open Design Alliance; not available in AutoCAD.
...	VBA	Current AutoCAD VBA code runs as-is in 32-bit BricsCAD for Windows; not available in BricsCAD Linux, Mac, 64-bit Windows, or Windows Standard
VSTA	...	VSTA is unavailable in BricsCAD.

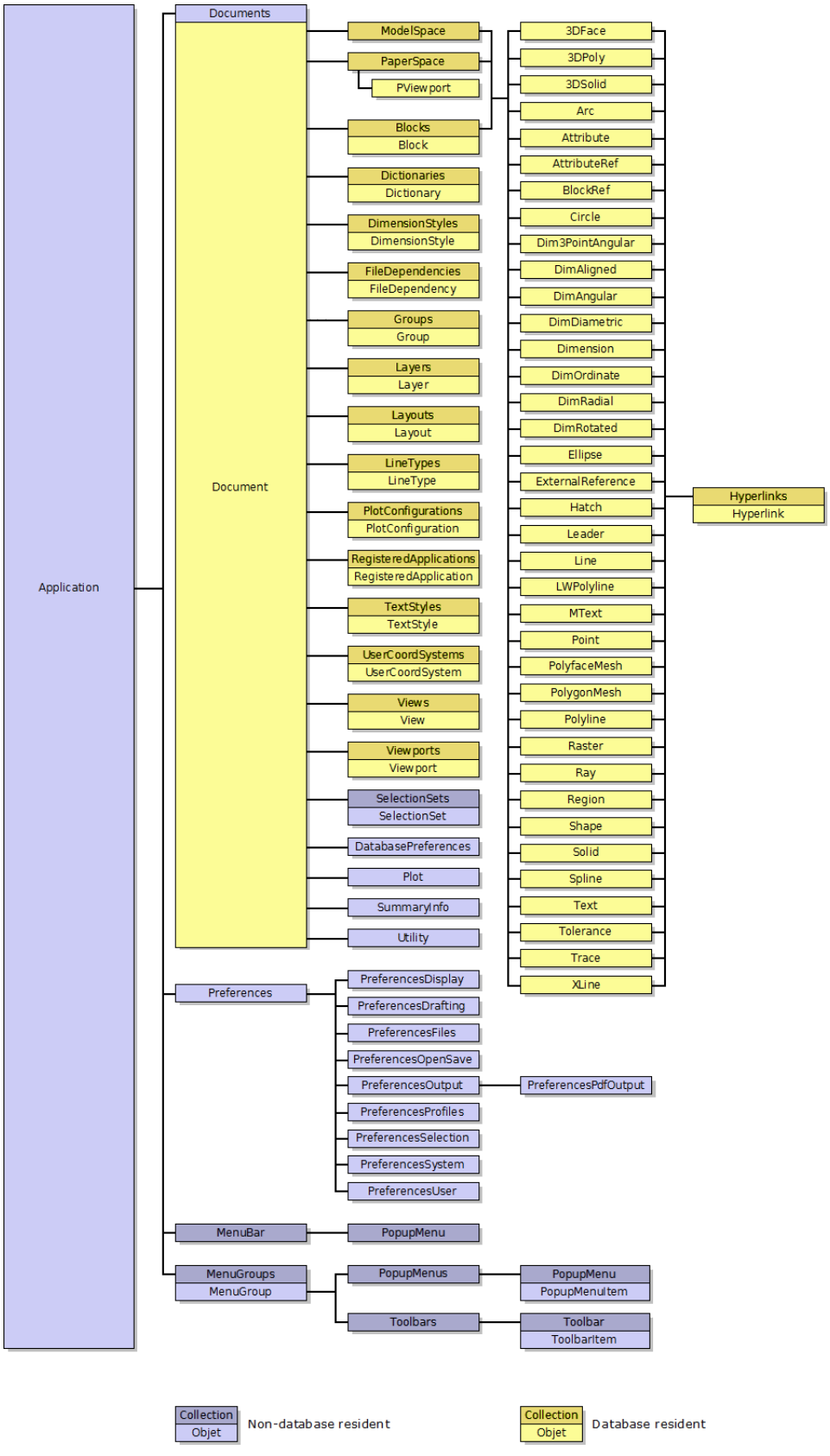
In general, BricsCAD provides a nearly identical subset of function names. In the case of non-compiled code, such as LISP and DCL, you just drop it into the BricsCAD environment. You recompile compiled code using headers provided by Bricsys. For writing C and C++ applications, BricsCAD offers BRX, which is code-compatible with AutoCAD's ARX. BricsCAD supports SDS, which is compatible with AutoCAD's ADS, although this API is deprecated by Autodesk and Bricsys.

You can reuse *.lsp* AutoLISP routines, and *.dcl* dialog control language files with no modification; in Windows only, *.dvb* projects (VBA macros). Detailed information is freely available from the Bricsys online developer reference at http://www.bricsys.com/bricscad/help/en_US/V17/DevRef.



BricsCAD V17 Automation Object Model

See https://bricsys.com/bricscad/help/ro_RO/CurVer/DevRef/source/COM_ComponentObjectModel_Diagram.htm



ABOUT BRX

BRX is 100% code compatible with ARX, AutoCAD's C++ interface. This means that you need only maintain one set of source code for both CAD platforms. They are not, however, *binary* compatible so modules compiled with ARX cannot be loaded directly into BricsCAD — and visa versa. First, recompile the source code, as follows:

BricsCAD compiles code and link with BRX to run on BricsCAD; The necessary **.h*, **.c*, and **.tlb* files are included in the BRX SDK

AutoCAD compiles code with ARX to run on AutoCAD

The BRX API was developed by Bricsys, and so is available for BricsCAD exclusively. The API is supported on BricsCAD V8 (or higher) Pro and Platinum only, not on BricsCAD Classic or releases prior to V8. The higher the BricsCAD version, the more BRX functions are supported. BRX offers the following functions in common with ARX.

This list is not exhaustive:

- Common basic functionality, such as AcRx, AcAp, AcCm, AcDb, AcEd, AcGe, AcGi, AcGs, and AcUt
- Multiple document interface using AcApDocument, AcApDocumentIterator, AcApDocManager, and so on
- Reactors like AcApDocManagerReactor, AcDbDatabaseReactor, and AcEditorReactor
- Custom objects derived from AcDbObject, AcDbEntity, and so on
- Transactions using AcDbTransactionManager, AcTransactionManager, and so on
- Input point processing with AcEdInputPointManager and AcEdInputPointMonitor
- MFC-based user interface extensions, such as AcUi and AdUi-based categories
- COM interfaces callable from C++
- Undocumented ARX functions, such as acdbSetDbmod, acedPostCommand, acedEvaluateLisp, ads_queueexpr, getCurrentPlotStyleName, and GetListOfPlotStyles
- Load on demand for commands registered through the AcadAppInfo interface
- Property palette inteface, OPM
- B-modeler code compatible with A-modeler
- Hidden Line and Brep APIs
- Managed wrapper classes for .NET API

TIP BricsCAD V17 is compiled with Visual Studio 2013 (platform toolset = v120). To be compatible, third-party C++ code with .dll extensions need to be compiled with the same V120 platform toolset.

ABOUT TX

The TX SDK produces TX modules files with the *.tx* extension, which are DLLs that are loaded at runtime by BricsCAD. BricsCAD is based on the Teigha libraries from Open Design Alliance, and so TX modules compiled with the TX SDK (Teigha eXtension software development kit) can be loaded to run in BricsCAD.

Prior to V12, the modules where named *.drx*. These cannot be loaded into V12 or later; you must recompile the source code using the latest TX SDK. TX classes, methods, and functions seem similar to those in ARX. There are, however, a number of differences:

- TX SDK enforces smart pointers in client code.
- Constructing and destructing objects are different from ARX.
- Control flow of error handling is different in ARX and TX applications, because error handling is based on exceptions thrown by the Teigha libraries, for the most part, and these need to be caught by the client code.
- TX SDK contains a subset of ARX, and so functions such as AcEdJig, AcApDocument, AcApDocManager, AcEdInputPointMonitor, and AcUi are missing.
- Some basic operations are done differently from ARX, such as retrieving the active database instance or opening entities.
- There are some minor differences in the class hierarchy of objects.

For more on how to use TX with BricsCAD, refer to the online documentation at http://www.bricsys.com/bricscad/help/en_US/V17/DevRef/source/TX_01.htm.

ABOUT .NET (WINDOWS ONLY)

The BricsCAD .NET API exposes the CAD system's functionality, and allows you to build managed code that runs under the .NET Common Language Runtime CLR. .NET is not available on Linux, Mac, or Classic versions of BricsCAD. With BricsCAD V15, the supported .NET runtime is version 4.0. See https://www.microsoft.com/net.

To set up a project with Visual Studio, create a class library using the class library wizard under your preferred .NET language. There are two DLLs that need to be referenced: *BrxMgd.dll* and *TD_Mgd.dll*. The optional *TD_MgdBrp.dll* handles the Brep APIs. These DLLs are located in the BricsCAD installation folder.

TIP When referencing these DLLs, it is important to set the Copy Local property to False. All other DLLs such as referenced COM DLLs or satellite DLLs, can have their Copy Local property to true, or as needed by your project. Samples projects are found in the *\Bricsys\BricsCAD\API\dotNet* folder.

PORTING AUTOLISP TO LISP

Most AutoLISP routines work directly in BricsCAD. Its LISP engine supports VL and VLA functions, and LISP reactors (except in the Linux and versions), as well as encrypted LISP; it does not support compiling to FAS (compiled LISP) files.

You may experience the following issues:

- BricsCAD's command line input can vary slightly from AutoCAD's. The solution is to verify the content of all (command) functions, or avoid using (command) altogether.
- BricsCAD does not implement a few AutoLISP functions. The solution is to rewrite the code, or to adapt external libraries.

DOSLib works with BricsCAD Pro and Platinum. It is a free library of LISP-callable functions not found in regular LISP. See http://wiki.mcneel.com/developer/doslib.

Porting DCL to BricsCAD

DCL routines work directly in BricsCAD for designing dialog boxes.

In addition, OpenDCL is fully supported and available for BricsCAD; see http://opendcl.com/wordpress.

Porting Diesel to BricsCAD

Diesel routines work directly in BricsCAD for macros and the status bar.

PORTING VBA TO BRICSCAD (WINDOWS ONLY)

AutoCAD and BricsCAD for Windows both use *.dvb* files for VBA projects. BricsCAD Pro and Platinum deliver VBA v7.1., and works both the 32- and 64-bit versions. VBA is not available in BricsCAD for Linux or Mac.

PORTING ADS TO SDS

Since ADS/SDS were developed nearly 20 years ago, Bricsys considers SDS *deprecated*, meaning developers should no longer use it. However, for backwards compatibility, Bricsys supports the old SDS interface.

(ADS is short for AutoCAD Development System, the first API for AutoCAD to use external libraries. SDS is short for SoftDesk Development System, a workalike first developed by SoftDesk for its IntelliCADD project.)

ADS code requires only a recompile using the BRX headers. To run an IntelliCAD-style SDS module on BricsCAD, the code must be adapted as described at http://www.bricsys.com/bricscad/help/en_US/V17/DevRef/source/SDS_01.htm.

PORTING COM TO BRICSCAD (WINDOWS ONLY)

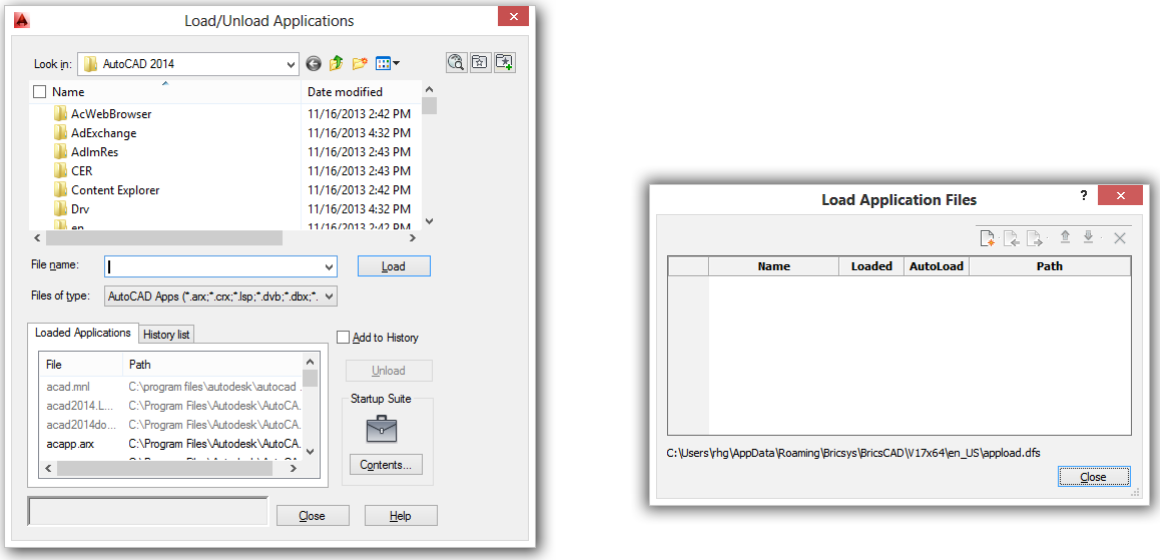
COM (Common Object Model) is available in Pro versions of BricsCAD, and is accessed through programming languages like VB, VBA, VB.NET, C, and C++.

Though BricsCAD's object model is quite similar to AutoCAD's, it is not identical. Nevertheless, most VBx code written for AutoCAD should work directly under BricsCAD. When you find a required element missing from the object model, the BricsCAD developer support team is open to creating the functions you require.

BricsCAD does not support VSTA (Visual Studio Tools for Applications).

LOADING APPLICATIONS INTO BRICSCAD

BricsCAD and AutoCAD use the **AppLoad** command to load applications into each CAD program. (NEW IN V17) The dialog box was redesigned.



***Left:** AutoCAD's application loader; **right:** BricsCAD's application loader*

Units

BricsCAD and AutoCAD share the same units conversion file, which is used by functions in LISP, SDS, and so on.

BricsCAD calls its file *default.unt,* while AutoCAD's file name is *acad.unt*.

THIRD-PARTY DEVELOPER SUPPORT

Bricsys notes that "There is day to day support for application developers who need assistance porting applications to BricsCAD, or simply require technical information about the porting process and the possibilities. The Bricsys development team has an extended section with dedicated developers for the different development environments (LISP, COM, ADS, ARX, .NET)."

When third-party developers request an addition to the API, it becomes a new feature in BricsCAD that end-users can employ. Bricsys does not charge third-party developers, unlike Autodesk. There is no fee to join, no annual membership, no charge for support, and no royalties on shipping products.

Visit https://www.bricsys.com/en_INTL/developers/ for more information.

CHAPTER FIVE

Operating Dual-CAD Design Offices

SOME FIRMS OPERATE BRICSCAD EXCLUSIVELY, BUT OTHERS RUN A MIX OF AUTOCAD AND other CAD systems, such as BricsCAD. This chapter explores the realities of running a dual-CAD shop, and explains how to solve issues that arise.

As well, we examine the benefits and drawbacks to running the Linux operating system as a cost-saving alternative to Windows or MacOS, and touch on the place of CAD on mobile devices and in Web browsers.

Why Use More Than One CAD System?

It's become common for design firms to license more than one brand of CAD package. Examples include AutoCAD and AutoCAD LT, AutoCAD and Solidworks, and AutoCAD and BricsCAD.

There is, however, more work involved in running CAD systems that are different. The differences lie in variations in capabilities, disparities in licensing policies, varying levels of hardware needs, and areas of incompatibility.

So why would a design firm cause itself apparently-unnecessary grief by taking on these problems? Dual-CAD firms tell me that they nevertheless adopt secondary CAD packages for these reasons:

- Cost savings
- Compatibility
- Capability

If I were a clever motivational speaker, I would call these "The Three Cs to Success."

LOWER TOTAL COST OF OWNERSHIP

For some firms it was too expensive in the past to pay $4,195 for every legal copy of AutoCAD. To save money, they run a majority of their seats on a lower-cost package, such as AutoCAD LT or BricsCAD.

For instance, a 100-seat design firm might split its workstations 10/90 between AutoCAD and the lower-cost software, saving the firm over $300,000 in initial licensing costs. The table below illustrates the dramatic savings that are possible right off the bat:

Number of Seats	Licensing Cost	Initial Savings
Pure AutoCAD		$ 0
100 of AutoCAD	$419,500 [1]	
Mix of AutoCAD and BricsCAD		$319,050
10 of AutoCAD	$ 41,950	
90 of BricsCAD Pro [2]	$ 58,500	
Pure BricsCAD		$354,500
100 of BricsCAD Pro	$ 65,000	

[1] AutoCAD is no longer available at this price (of a perpetual license), but I include it to make the arthemetic simpler.

[2] I chose BricsCAD Pro rather than Platinum, because the additional functions provided by Platinum are not found in AutoCAD, such as 3D constraints and assemblies.

The actual cost to license one hundred seats would be lower than shown by the table, because just about any CAD vendor offers customers better pricing on bulk purchases and network licenses.

Upgrades. Following the initial licensing cost, design firms can choose to spend on additional charges typically associated with software use:

- Upgrade fees
- Annual maintenance or support fees (include upgrades at no added cost)

Autodesk as of January 31, 2016 eliminated AutoCAD upgrades and perpetual licenses all together. In this regard, BricsCAD also has the purchasing advantage over AutoCAD. BricsCAD allows you to upgrade your perpetual license of BricsCAD at any time in the future for US$250/license.

Subscriptions. The third alternative is to purchase subscriptions instead of perpetual licences. At Payments are made upfront to Autodesk for a month-long use of the CAD software, or else upfront for one year, two years, or three years. This corresponds to being billed monthly, annually, biannually, or triennially. Bricsys offers only annual subscriptions.

The table shows the cost for subscribing to 100 licenses on one-year plans, the only length com mon to both CAD programs. No mixing allowed! To be on subscription, Bricsys requires all seats at a single site be on subscription; Autodesk offers only subscription pricing.

Number of Seats	Annual Subscription Price [1]	Annual Cost Savings
Pure AutoCAD		
100 of AutoCAD	$140,000	$ 0
Mix of AutoCAD and BricsCAD		
10 of AutoCAD	$ 14,000	
90 of BricsCAD	$ 22,500	$ 103,500
Pure BricsCAD		
100 of BricsCAD	$ 25,000	$ 115,000

[1] Prices in US$ as reported by each vendor's Web site on 12 November 2016

The advantages and disadvantages to paying by subscription are as follows:

- **Pro:** The upfront financial cost is 1/3 less than that of a perpetual license
- **Con:** A subscription becomes more expensive than a pure perpetual license after 3 years

- **Pro:** Being a subscription cost, the amount is 100% deductible from income taxes annually
- **Con:** In some jurisdictions, permanent licenses are fully depreciated in just two years

- **Pro:** Firms can reduce their cost by reducing their license count when the work load lessens during recessions
- **Pro:** Firms can rent software monthly for the workload jumps
- **Con:** Firms may be pressured by CAD vendors to not reduce their license count under the threat of higher fees (as occurred during the 2008 recession)
- **Con:** Subscription-paid software stops working after 15 to 30 days, should the firm be unable to afford the next payment

- **Pro:** Subscriptions often include additional benefits, such as free upgrades, better support, and extra software at no cost
- **Con:** Subscription prices and benefits fluctuate as CAD vendors alternate between wanting more revenue (prices go up in the long term) and wanting more new customers (subscription prices go "on sale" or benefits increase in the short term)

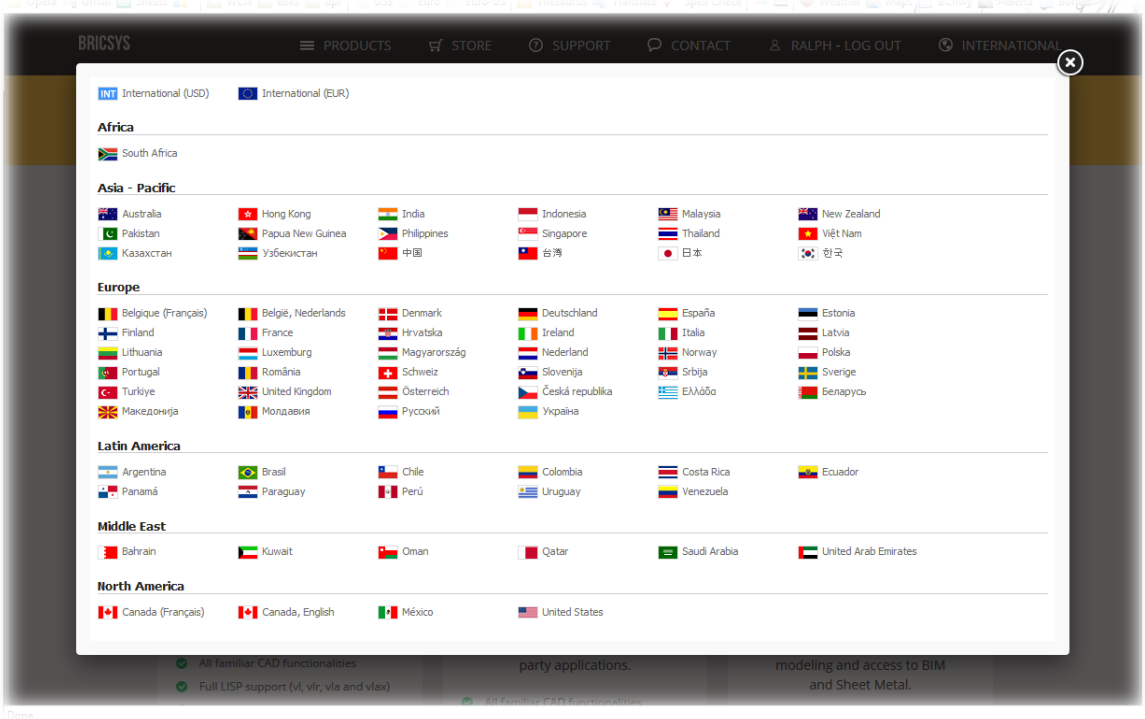
Hardware. I did not include the benefit of using older and slower hardware with BricsCAD, as this cannot be easily quantified financially; there are too many variations in workstations and pricing. BricsCAD does not need the more expensive computers and graphics boards that AutoCAD requires to run well.

Nevertheless, the advantage goes to BricsCAD, as initial hardware costs are lower and subsequent hardware upgrades are rarer.

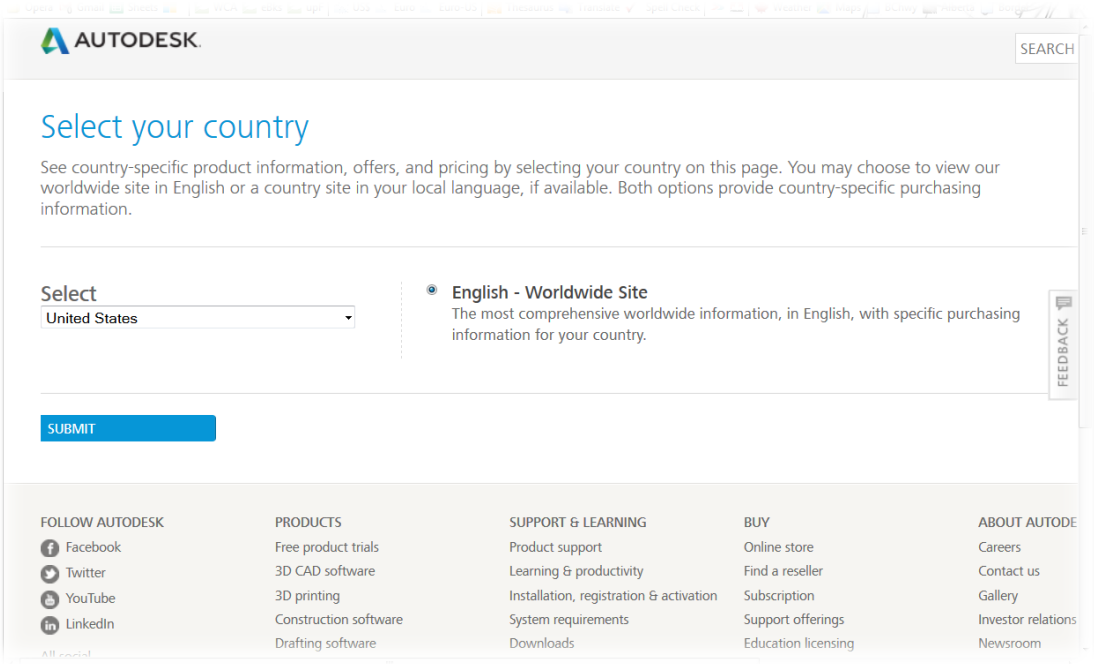
Country-Biased Pricing

Both Autodesk and Bricsys charge different prices for different countries. You can learn the current price schedule for your country by visiting these online shops:

- Autodesk: http://www.autodesk.com/store
- Bricsys: https://www.bricsys.com/estore/



Choosing an international location from the online store at Bricsys.com



Choosing an international location from the online store at Autodesk.com

The pricing situation is acute for firms in developing countries, where starting architects make as little as $300 a month. In my opinion, I find it disturbing when software companies charge more in these high-growth, low-income countries, thereby placing software tools out of reach for many potential customers. Ironically, software companies complain about the high rate of piracy in developing countries — they fail to see the connection.

The non-democratic pricing model puts ethical design firms in a bind. They cannot afford a full house of expensive CAD software licenses, yet they need to show large clients that they are running a clean shop with no pirated software. There is a solution.

Solutions to High License Fees

For firms that cannot afford Western prices, the solution is to license lower-cost products, specifically AutoCAD LT, BricsCAD, and the like. Indeed, BricsCAD Classic provides design firms with a more-capable CAD package at half the price of AutoCAD LT.

Another way to save money is to run the free Linux operating system on computers, instead of the pricier Windows. (MacOS is free, but runs only on Apple-branded computers that tend to be the most expensive kind of computer.) The catch to offices employing Linux is that the CAD vendor must have a version of the software that runs on Linux. Bricsys does; Autodesk does not.

Linux is doubly cost-effective, because it runs well on older, less powerful computers. Newer releases of Windows typically require new hardware, if only because the updated operating system no longer supports older device drivers or software.

In summary, BricsCAD is triply cost-effective:

- BricsCAD Platinum is priced 4x less than AutoCAD, and 1.5x less than AutoCAD LT
- BricsCAD runs on Linux, which is free
- BricsCAD and Linux have lower hardware demands than AutoCAD and Windows, and so run effectively on older computers

MAXIMIZING COMPATIBILITY

Like all responsible capitalist corporations, design firms look to reduce their expenses, and so prefer the lowest-cost system that produces the highest profits with the fewest expenses — measurable and unmeasurable. I listed some of the measurable expenses above.

The #1 unmeasurable expense comes from difficulty in using a software system. In the case of CAD, this can mean difficulty of the user interface, links to external programs, and absolute compatibility with the industry standard, AutoCAD. For this last reason, design shops employ at least a few seats of AutoCAD.

Autodesk tries to make sure that AutoCAD stays ahead of the competition, whether through technology or through marketing. For example in marketing, when in the mid-1990s IntelliCAD began threatening sales of the 10x more expensive AutoCAD, Autodesk launched a campaign that effectively warned customers away from the upstart. (In the campaign, Autodesk claimed that AutoCAD LT was the only low-priced CAD package that was 100% DWG-compatible with AutoCAD. The problem with the claim at the time was that it was not entirely accurate, for AutoCAD LT in those days could not deal with all the entities created by AutoCAD.)

For many years, Autodesk put huge resources into leap-frogging AutoCAD ahead of the competition, making the "100% Pure DWG" situation even more true. In recent years, however, Autodesk has slowed its pace, curiously enough; as of AutoCAD 2014 onwards, the flagship software has gained only a few new functions each year, and the file format remaining unchanged for more five years, as of this writing.

Nevertheless, most design firms have at least one license of AutoCAD on the chance that drawings from clients might not reproduce correctly in IntelliCAD or BricsCAD. This is like firms saving money by standardizing on the free Libre Office package, yet maintaining a license of Microsoft Office to ensure compatibility with files created by the *de facto* standard in office software.

Open Design Alliance. The industry counterweight to Autodesk is the Open Design Alliance. The ODA was established in the late 1990s to document Autodesk's DWG format, which has been kept proprietary through a lack of documentation. Today, the organization has 1,200 members and provides APIs that allow members' software to read and write AutoCAD DWG, MircoStation DGN, Adobe PDF files, and other popular file formats, such as Revit RVT files. The organization also

provides other resources, such as an equivalent to the ARx programming interface, ADT and MDT object enablers, and licensing of add-on software like ACIS and C3D solid modeling kernels. http://www.opendesign.com

ODA and its contract programmers do the hard work by figuring out what's inside DWG. This means that BricsCAD, IntelliCAD, and other firms can concentrate on adding features to their CAD systems. The bad news is that the *content* of the DWG file changes every year as Autodesk adds more capabilities and object types to AutoCAD. The qualified good news is that Autodesk tends to freeze the *format* for at least three years at a time. ODA's programmers usually figure out the new content in under six months.

I noted earlier that Autodesk would not document DWG for many years, despite calling it the worldwide standard for engineering drawings. As a result of the formation of the ODA, Autodesk relented and documented DWG through its own API, RealDWG.

CAPABILITY

Autodesk for a few years added really big features to AutoCAD, such as 3D mesh modeling, 3D surfaces, point cloud processing, and a new rendering engine. The bad news is that these huge additions kept workalikes from replicating all these complex functions in their entirety; on their own, they don't have the programming resources. Banded together under ODA, however, they make progress.

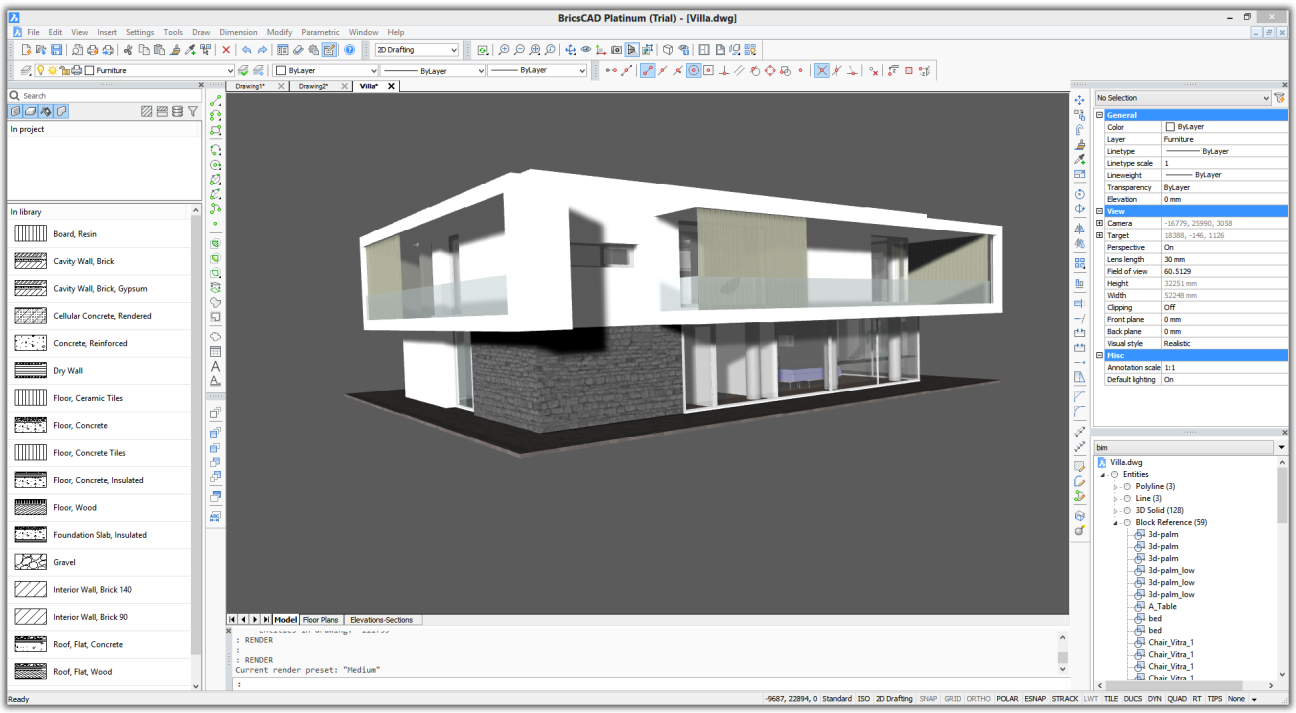
The good news for workalikes is that there is no need to replicate AutoCAD completely. It turns out that 3D point clouds and such are of little interest to heads-down drafters. If a design firm needs the capability, there are many third-party stand-alone products that do as good a job as AutoCAD — or better.

The majority of AutoCAD and BricsCAD users produce 2D drawings. A Solidworks product manager proclaimed at a users conference that "2D will go on and on, probably for 50 years." At any user conference, the biggest cheers are reserved for new functions that save time in 2D drafting, like automatic balloon placement. Even in hard-core 3D CAD environments, such as Catia, the numbers indicate that more than 50% of drawings are produced in 2D. (Catia, from Dassault Systemes, is high-end 3D modeling software used by aircraft and automotive firms, among others.)

Nevertheless, 3D cannot be ignored, and workalikes traditionally have been weak in that: IntelliCAD and other workalikes achieve today what AutoCAD did more than a decade ago. The primary exception is BricsCAD, which is taking giant strides in beefing up its 3D offerings.

- With V11, Bricsys added a higher-priced Platinum Edition that offered 3D history-based parametric modeling, known as X-Solids. It included a parametric parts library, called X-Hardware.
- With V12, Bricsys added 3D direct modeling and 2D constraints to all editions, with 3D constraints added to the Platinum Edition.
- With V13, Bricsys added assembly modeling for linking two or more 3D models using constraints, kinematic analysis for checking motion and interference between parts, and bills of materials.

- With Communicator, Bricsys added import and export for popular MCAD formats such as Solidworks, Inventor, and IGES
- With V14, Bricsys added sheet metal design and assemblies.
- With V15, Bricsys greatly expanded sheet metal design, began on BIM (building information modeling for architects), and added a link to CAM.
- With V16, Bricsys added 3D surfacing, beefed up the capabilities of BIM and generative drafting, added 3D lofting, and began importing MCAD assemblies.
- With V17, Bricsys added 3D compare of modified 3D models, IFC certification, and real-world material specifications.



BricsCAD showing BIM material definitions (panel on the right)

BricsCAD is on its way to becoming something like an AutoCAD-compatible version of Inventor or Revit, but for under $2,000.

Running BricsCAD & AutoCAD in One Office

To run more than one CAD system in your office successfully, it is crucial that you understand the differences between them. The differences exist, because the abilities of AutoCAD and BricsCAD differ.

To implement a dual-OS office, this ebook is your primary reference; as well, you may find it helpful also to refer to my *Alphabetical Command Reference for AutoCAD 2013-2014* (http://www.worldcadaccess.com/ebooksonline/2015/07/acr.html), which lists all AutoCAD commands in alphabetical order, along with options and all the ways of launching each command. It is valid for nearly all versions of AutoCAD since Release 12, as it includes an update history.

Next, you should establish an in-house workflow to assign drafting tasks appropriate to each CAD system. I describe this in the following section.

When you find a feature missing, then you will need to find a workaround. For instance, when BricsCAD cannot handle certain entities, you can xref drawings from AutoCAD. BricsCAD can display nearly anything that AutoCAD can draw, but does not create or edit every entity type.

In summary, BricsCAD has the following capabilities *vis a vis* AutoCAD:

Activity	BricsCAD can...
View	...display nearly all AutoCAD entity types, even if it cannot edit or create them
Edit	...edit most AutoCAD entities, although sometimes only through the Properties pane
Create	...create many AutoCAD entities, but fewer than it can edit

See chapter 3, "Drawing File Compatibility," for the nitty gritty details on each and every DWG object.

DIVIDING WORKFLOWS BETWEEN AUTOCAD & BRICSCAD

You probably are well acquainted with the *workflow* in your office, the route that drawings take through the office — typically from the general to the specific. For instance, one of my clients has the following workflow:

1. Receive DWG drawing files from architects
2. Review the dimensions on received drawings for dimensional accuracy
3. Create overall elevation views of the building's faces; make plan views of each floor
4. Draw up assembly drawings for fabricators
5. Make detail drawings of every item, and then generate bills of materials
6. Plot drawings on B- or C-size paper
7. Send completed paper drawing sets to clients and fabrication shops

As much as possible, the work is done in BricsCAD, because it operates on the majority of workstations. The only work handled by AutoCAD are design functions BricsCAD is unable to complete.

This particular design firm took the time to list the CAD functions they employed in their office, and then created two lists: (a) features that work in both BricsCAD and AutoCAD and (b) those that work only in AutoCAD.

Here is an example of the lists they created. The firm note the *usefulness* of features to their workflow. These lists are not exhaustive, but specific to the needs of this particular design firm.

First, features common to both CAD systems:

Features that Work in BricsCAD and AutoCAD	Level of Usefulness
Template DWT files	Very useful for speeding up initial drawing creation
Field text	Very useful for automating text
Data extraction and spreadsheets	Very useful
Hyperlink command	Very useful for linking to other drawings
Geometric and dimensional constraints	Very useful; using dimensional constraints for sizing objects
Sheet sets	Very useful for organizing groups of drawings
Mleaders, editing, styles	Very useful for joining multiple leaders into one; and for lining up leaders neatly
Overkill	Useful for cleaning up drawings
LISP / AutoLISP	Useful for automating some routine drafting
CUI / Customization	Useful in some aspects, such as combining commands
Explorer / DesignCenter, Tool Palettes	Probably useful for sharing and accessing content
Drawing Views	Probably useful for generating 2D plans from 3D; firm had not yet deployed this function
Annotative scaling	Not useful

And here is the usefulness of functions found only in AutoCAD (not BricsCAD):

Features Specific to AutoCAD	Level of Usefulness
DimBreak, DimSpace, DimJogLine	Very useful for editing dimensions
LayTrans command	Useful for bulk editing layer names of incoming drawings
Dynamic blocks	Useful for creating complex linetypes
Measure and Divide	Useful for placing QDim dimensions; BricsCAD lacks QDim
QDim	Useful when used with Measure
Check Standards commands, DWS files	Too limited in scope to be useful
Active Recorder	Not useful
Point cloud processing	Not useful
3D mesh and surface modeling	Not useful

Your designation of useful and useless functions may differ. Concentrate on dealing with functions that are useful in the workflow; useless and limited functions can be ignored. With each release, the lists must be updated as new functions are added to both CAD systems.

STRATEGIC IMPLEMENTATION

While several employees may be keen to implement more efficient drafting methods on BricsCAD and AutoCAD, it pays to place *one* strategic employee in charge of CAD management and training for everyone.

Here is the plan that one design firm arrived at:

- Decide on the split between the Classic, Pro, and Platinum versions of BricsCAD
- Upgrade all Linux, Mac, Windows licenses of BricsCAD to the latest version
- Determine a split of drafting tasks between AutoCAD and BricsCAD, recognizing the limits of BricsCAD
- Automate 2D drafting processes as much as possible
- Introduce a few seats of Inventor for handling specific 3D constructions, as well as forms of automated drafting of which AutoCAD and BricsCAD are incapable

TIP Inventor licenses include AutoCAD free.

- Create a steering group to ensure the new techniques are disseminated throughout the firm; ensure progress is made
- Consider hiring local trainers for specific topics; create a CAD programmer position
- Review the implementation in a year's time

COMMON OPERATIONS THROUGH FILE PATHS

BricsCAD and AutoCAD drawings employ many support files. Examples include linetype definitions, font files, and external references. , The good news is that most of them are the same, and so the two CAD systems can share the same support files, reducing mangement complexity. The only catch is that Autodesk names many support files *acad.**, while Bricsys uses *default.**; these files can be renamed.

To keep track of them logically, CAD vendors store support files in specific folders. Both CAD programs let you specify paths to these folders.

BricsCAD specifies paths in the **Settings** dialog box: see the **Program Options** section
AutoCAD specifies paths in the **Options** dialog box: see the **Files** tab

In older, simpler times, all support files were stored in a folder named *Support*. But as Microsoft made Windows more complex, it required software makers to scatter support files into many folders for those cases when Windows computers are used by more than one user.

Local files are stored on the computer you use; these are files specific to each user and each program, such as DWG drawing files and local customization files.
~~**LocalLow** files are stored like Local files, but with a lower integrity level; used by Web browsers when Windows protected mode is on. BricsCAD and AutoCAD do not use LocalLow folders.~~

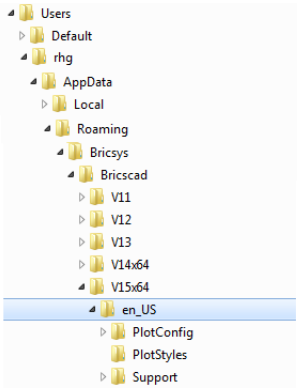
Common files are stored on the computer you use; these are files, such as font files and printer drivers, that are common to many programs. CAD programs make use of these files.

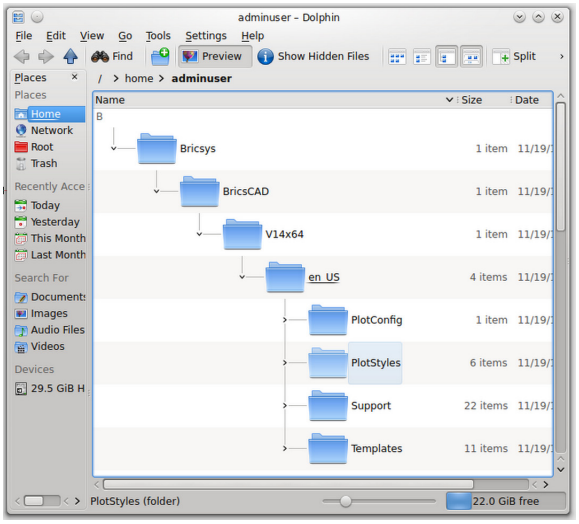
Temporary files are stored "anywhere," locally or on the network; these files are created by CAD programs for the duration of the editing session, such as automatic backup files.

Roaming files are stored on any computer; these files are specific to you, such as customized linetype and hatch pattern files, and so are accessible from any networked computer. See Roamable Profiles later in this chapter.

Network files are stored on the network and are accessible to everyone, such as blocks and template files.

In Windows, these folders are usually found in a hidden folder named "AppData" under *C:\users\<login>*, where "<login>" is the name by which you log into Windows. My login name is *rhg*, and so all of my Local, and Roaming folders are found under *C:\users\rhg\AppData*.





Left: *Local and Roaming support folders in Windows*
Right: *Support folders in Linux*

To maintain compatibility with Windows, BricsCAD for Linux uses similar folder names and structures, although without the Local and Roaming folders. All support folders are found in this path:

```
/home/<login>/Bricsys/BricsCAD/V17
```

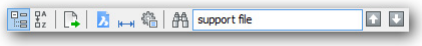
TIP BricsCAD provides users with the following commands to make it easier to handle support files:
SupportFolder opens the *C:\Users\<login>\AppData\Roaming\Bricsys\BricsCAD\V17x64\en_US\Support* folder.
TemplateFolder opens the *C:\Users\<login>\AppData\Local\Bricsys\BricsCAD\V17x64\en_US\Templates* folder.
WhoHas display ownership information for a selected drawing file.

Tutorial: How to Add AutoCAD's Support Folders to BricsCAD

If AutoCAD is installed on the same computer as BricsCAD, then you can point BricsCAD's support paths to AutoCAD's folders. This allows you to use common standards for both programs, such as hatch patterns, linetypes, and fonts in common.

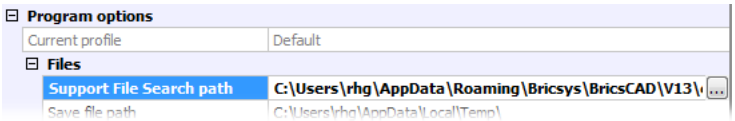
To direct BricsCAD to use AutoCAD's support files, open the Settings dialog box, and then access the **Program Options**, like this:

1. Start BricsCAD, and then enter the **Settings** command.
2. In the Search field, enter **support file**.



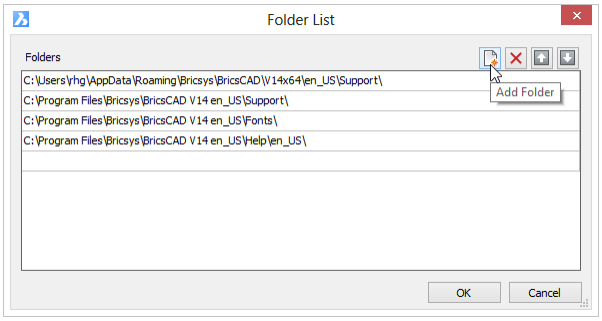
Searching for the phrase "support file"

3. Notice that the Settings dialog box jumps to the Support File Search Path item.



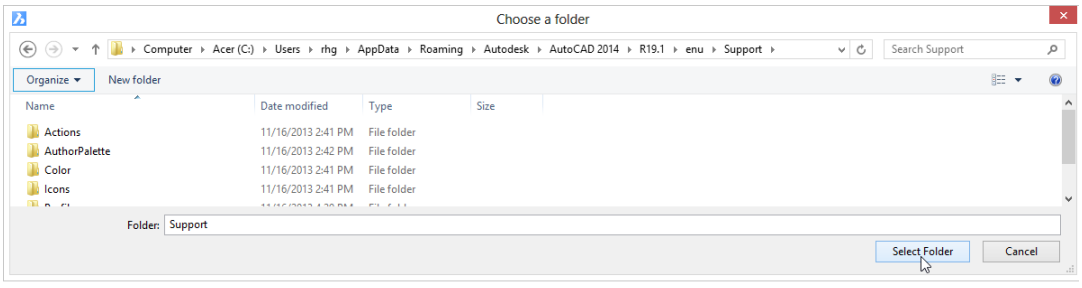
The support file search path entry in the Settings dialog box

4. Click the … **Browse** button. Notice the Folders List dialog box.



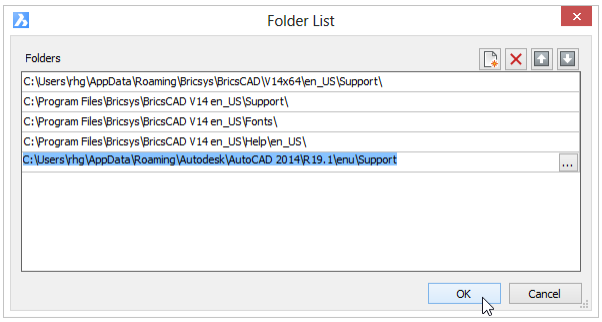
Adding folders to BricsCAD's search path

5. In the Folders List dialog box, click **Add Folder**.
6. To look for the folders you want to add, click … **Browse**.
7. In the Choose a Folder dialog box, navigate to the AutoCAD folder you wish to add, and then click **OK**.



Selecting a folder to add to the search path

Notice that the folder is added to the list. BricsCAD highlights the folder to indicate it is newly added.



New folder added to the search path

8. Repeat the process to add the locations of other support folders, such as:
   DWT drawing template files at *C:\Users\<login>\AppData\Local\Autodesk\AutoCAD 2017 - English\R20.0\enu\Template*
   Most other support files at *C:\Users\<login>\AppData\Roaming\Autodesk\AutoCAD 2017 - English\R20.0\enu\Support*
9. When done, click **OK**.

TIP You can do the same process in AutoCAD: use its CUI dialog box's Files tab to point AutoCAD to BricsCAD support folders.

USER PROFILES

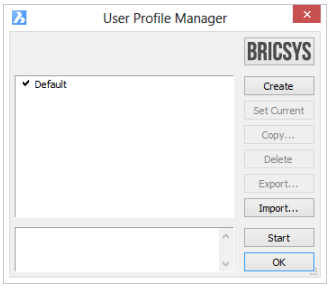
BricsCAD and AutoCAD both support *user profiles* to store each user's customization settings. After changing settings with the BricsCAD **Settings** and AutoCAD **Options** commands, you save the all the settings in a *.arg* user profile file. The idea here is that you can make multiple profiles that customize each CAD program for different users or for specific projects.

Profiles are made differently in each CAD package:

BricsCAD creates user profiles through an external application, *UserProfileManager.exe*.
AutoCAD creates user profiles through the Profiles tab of the Options dialog box.

To access BricsCAD's UserProfileManager program:

- Click the Windows 7 **Start** button, and then choose **All Programs | Bricsys | BricsCAD V17 | User Profile Manager**
- In Windows 8 and 10, press **Windows+Q** and then enter "user profile manager" in the **Search** field
- Or access it from inside BricsCAD by entering the **ProfileManger** command



User Profile Manager is a stand-alone program with BricsCAD

To save the current user interface configuration, click **Create** and then give the profile a name.

To switch to another profile, choose it from the list, and then click **Set Current**.

To read an *.arg* file from AutoCAD, click **Import**.

Launching BricsCAD with a User Profile

To launch BricsCAD with a named user profile, add the **/p** switch to its desktop shortcut's properties:

1. To access the properties, right-click the BricsCAD shortcut icon on the desktop, and then choose **Properties** from the shortcut menu.
2. Edit the **Target** field to look like this (changes shown in blue):

   ```
   "C:\Program Files\Bricsys\BricsCAD V17\bricscad.exe" /P <UserProfileName>
   ```

   For example, replace <*UserProfileName*> with the *.arg* file's name, such as *myprofile.arg*:

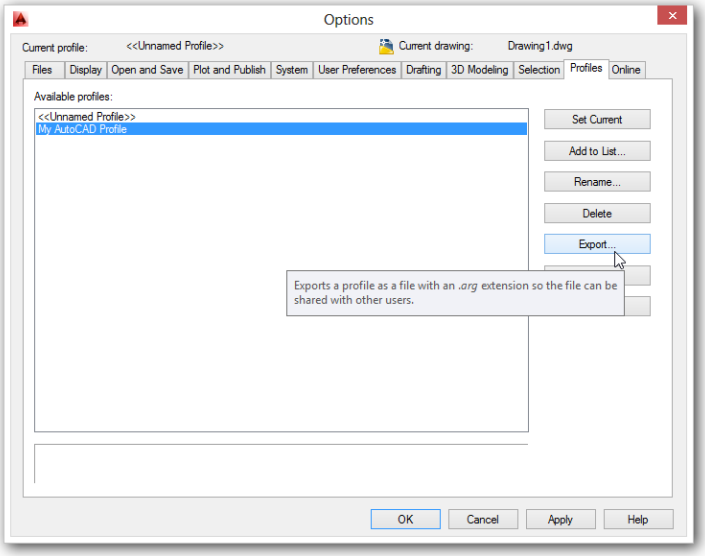
   ```
   "C:\Program Files\Bricsys\BricsCAD V17\bricscad.exe" /P myprofile.arg
   ```

Tutorial: How to Import AutoCAD Profiles into to BricsCAD

Both programs use the same format for *.arg* files, and so you can import AutoCAD-generated profiles into BricsCAD. Follow these steps to export and import them.

Firstly, export the *.arg* file from AutoCAD:

1. In AutoCAD, enter the **Options** command, and then click on the **Profiles** tab. (See figure above.)



Exporting a user profile from AutoCAD

2. Choose a profile from the list, and then click **Export**.
3. Select the folder into which profile file should be saved. If you wish, change the file name.
4. Click **Save**.
5. Click **OK** to exit the dialog box.

Secondly, import the *.arg* file to BricsCAD:

1. In BricsCAD, from the **Tools** menu, choose **User Profile Manager**.
2. In the User Profile Manager, click **Import**.
3. Choose the *.arg* file exported from AutoCAD, and then click **Open**.
4. To apply the profile, click **Set Current**.
5. Click **OK** to exit the program.

ROAMING PROFILES

BricsCAD and AutoCAD both support *roaming profiles*, which let you "roam" about the office and use the CAD program on any computer connected to the office network. Your profile is identified automatically by the login name you entered when you access the computer. The benefit is that BricsCAD and AutoCAD are customized automatically with your settings.

Not all CAD files are roamable; some remain local, such as DWT template files. This is why roaming and non-roaming (local) files are kept in separate folders. It is up to the software maker to decide which are which.

AutoCAD Support Folders

AutoCAD's nonroamable (local) files are in *C:\Users\<login>\AppData**Local**\Autodesk\AutoCAD \R20.0\enu* and consist of the following files:

- Template files (DWT, DST, DGN)
- Web Services

AutoCAD's roamable files are in *C:\Users\<login>\AppData**Roaming**\Autodesk\AutoCAD \R20.0\ enu* and consist of the following files:

- Data links
- Language packs
- Migration
- Plot styles (CTB, STB), plotter parameters (PMP), and plotter configurations (PC3)
- Support files (CUIX, FMP, LIN, MLN, MNL, PAT, PGP, PSF, UNT, and so on)

BricsCAD Support Folders

BricsCAD's nonroamable (local) files consist of the following ones:

- Template files (DWT)

The files are found by following these OS-specific paths:

Windows	*C:\Users\<login>\AppData**Local**\Bricsys\BricsCAD\V17x64\en_US*
Mac	*/users/<login>/Library/Prreferences/Bricssys/BricsCADV17x64/en_US/*
Linux	*home/<login>/Bricsys/BricsCAD/V17x64/en_US/*

BricsCAD's roamable files consist of the following ones:

- Plot styles (CTB, STB), and plotter configurations (PC3)
- Support files (CUI, FMP, LIN, PAT, PGP, PSF, UNT, and TXT)

The files are found by following these OS-specific paths:

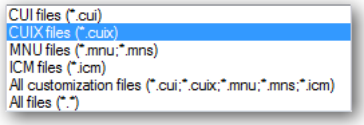
Windows	*C:\Users\<login>\AppData**Roaming**\Bricsys\BricsCAD\V17x64\en_US*
Mac	*/users/<login>/Library/Prreferences/Bricssys/BricsCADV17x64/en_US/*
Linux	*home/<login>/Bricsys/BricsCAD/V17x64/en_US/*

TIP You can change in BricsCAD the path to local and roamable folders with system variables **LocalRootPrefix** and **RoamableRootPrefix**. This is useful when the content of the folders is stored on a central server.

Tutorial: Importing Menus Files from AutoCAD

If you have menus that you customized in AutoCAD, then you probably can use them in BricsCAD. Follow these steps to import menu files from AutoCAD:

1. Use the **Customize** command to open the Customize dialog box.
2. At the right end of **Main Customization File** field, click the ... button.
3. In the Select Main CUI File dialog box, click the **Files of Type** droplist. Notice the list of file types:



Selecting a menu file type to import

   - **CUIX** — compressed CUI files that also store resources, like icon files; in use by AutoCAD since release 2012 and by BricsCAD since V14
   - **CUI** — standard menu files used by AutoCAD since release 2007 and by BricsCAD since V8
   - **MNU** or **MNS** — legacy menu and support files used by AutoCAD and by AutoCAD LT prior to release 2007
   - **ICM** — IntelliCAD menu files used by BricsCAD prior to V8 and by IntelliCAD-based systems

4. Choose a file type, select a file name, and then click **OK**. Notice that the menu structure changes to match the newly-imported file.

Careful! Although BricsCAD imports AutoCAD menu files effortlessly, menu actions sometimes do not work, because AutoCAD macros can contain macro code or metacharacters not supported by BricsCAD.

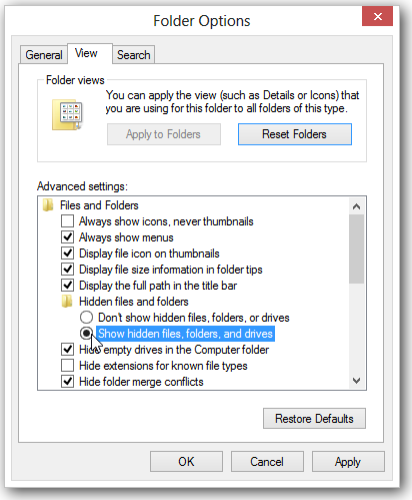
Tutorial: Making Hidden Folders Visible in Windows

Local and roaming folders can be difficult to find, because unfortunately they are typically hidden by Windows and MacOS. (They are not hidden in Linux.) Because I access them frequently, I unhide the folders.

TIP If you find yourself accessing these folders often, create shortcuts on your computer's desktop to them. Here's how: hold down the **Ctrl+Alt** key while dragging the folder name from Explorer onto the desktop.

Here is how I do this in Windows:

1. First, make *all* hidden folders visible by following these steps:
   a. In Windows, open File Explorer, and then choose Options:
      Windows 7: from the Tools **menu**, choose **Folder Options**.
      Windows 8 and 10: choose the **View** tab, and then from the Show/Hide panel, click **Options**.
   b. In the dialog box, choose the **View** tab,
   c. Under Advanced Settings, turn on **Show Hidden Files and Folders**.



Accessing the option to reveal hidden folders

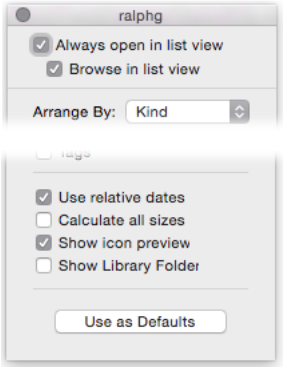
2. Now that hidden folders are visible, follow these steps in Explorer:
   a. Go to the *C:\users\<login>\appdata* folder.
   b. Right-click the folder, and then choose **Properties**.
   c. Uncheck **Hidden**, and then click **OK** to close the dialog box.

You can now see the Local and Roaming folders.

Tutorial: Making Hidden Folders Visible in MacOS

The Library folder is where BricsCAD stores its support files on Mac computers. Here is how to reveal the folder in MacOS:

1. Open Finder, and then navigate to your user folder. In my case, it is "ralphg."
2. From the **View** menu, choose **View Options**.
3. In the dialog box, notice that the **Show Library Folder** option is turned off. Click it to turn it on.



Unhiding hidden folders in MacOS

4. Close the dialog box. Notice that the Library folder is now visible.

Tutorial: Loading AutoCAD's PGP File into BricsCAD

The PGP file holds alias abbreviations for command names. If you have customized aliases in AutoCAD, then you can use them in BricsCAD. Here is how to load the PGP file from AutoCAD into BricsCAD:

1. Use Windows Explorer to copy the *acad.pgp* file **from** this folder:

   C:\Users\<login>\AppData\Roaming\Autodesk\AutoCAD \R20.0\enu\Support
2. Rename it *default.pgp*.
3. Place the renamed file in this BricsCAD folder:
   - **Windows** *C:\Users\<login>\AppData\Roaming\Bricsys\BricsCAD\V17x64\en_US\Support*
   - **Mac** */Users/<login>/Library/Preferences/Bricsys/BricsCAD/V17x64/en_US/Support*
   - **Linux** *home/<login>/Bricsys/BricsCAD/V17x64/en_US/support*

   (Remember to replace <login> with your Windows login name.)

It turns out that in BricsCAD you cannot simply use the Customize dialog box's **Program Parameter File** field, because it does not allow you to enter a different path.

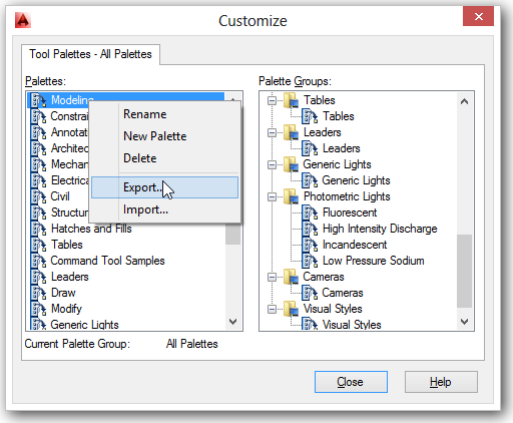
TIP To transfer files from a Windows computer to a Mac or Linux computer, use a USB thumbdrive or a file transfer service like Dropbox.

Tutorial: How to Export AutoCAD Palettes to BricsCAD

If you have customized the content of AutoCAD's Tools Palette, then you can use them in BricsCAD, because they use the same *.xtp* file format for exporting and importing palettes. XTP is short for "xml tool palettes," and is a file format based on XML, a self-documenting version of HTML that is often used in data exchange situations.

To import palette files from AutoCAD to BricsCAD, follow these steps:

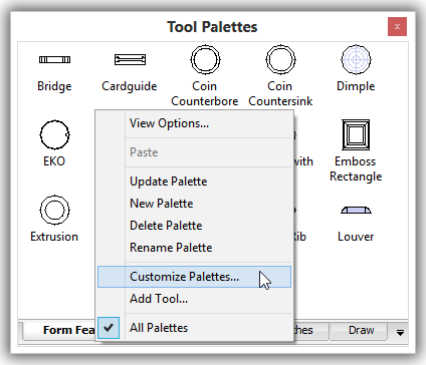
1. Start AutoCAD, and then enter the **Customize** command.
2. In the Customize dialog box, right-click the palette you want to export. From the shortcut menu, choose **Export**.



Choosing palettes to export from AutoCAD

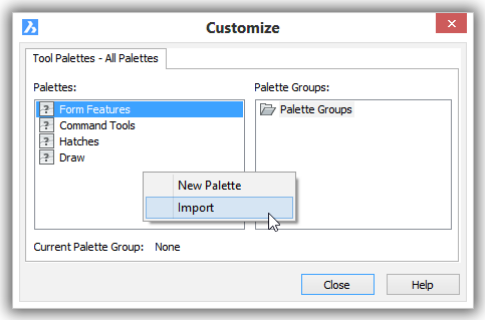
3. In the Export Palettes dialog box, choose the folder in which to place the exported XTP file, and then click **Save**. (I tend to use the Desktop, because it is easy to find later!)

4. Switch to BricsCAD.
5. Right-click the Tools Palette bar, and then choose **Customize Palettes**.



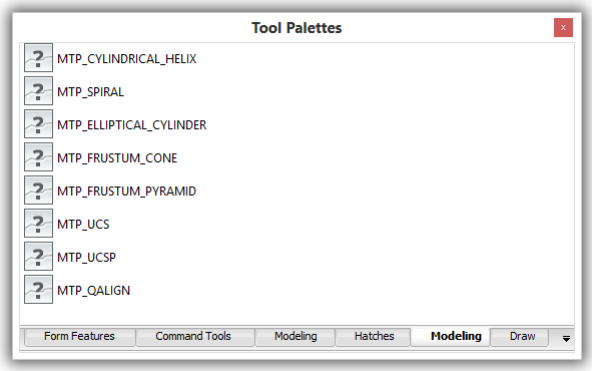
Accessing the Customize dialog box in BricsCAD

6. In the Customize dialog box, right-click any palette, and then choose **Import** from the shortcut menu.



Importing .xtp files into BricsCAD

7. In the Import Palettes dialog box, choose the XTP file you exported from AutoCAD, and then click **Open**. Notice that it is added to the list of Palettes.
8. Click **Close**. Notice that the Tool Palettes bar now has a new tab named after the palette you imported. The icons will probably consist of ?, because the icon files are unavailable.



Icons missing from imported AutoCAD tools palette

9. Click an icon; notice that the command (probably) works!

(NEW IN V17) BricsCAD supports groups of palettes.

The Dual OS Office

To further save money, some firms switch some of their workstations from Windows to Linux. One firm told me that replacing Windows with the free Linux operating system saves them 10% of their annual IT budget.

AutoCAD is not available for Linux, but BricsCAD is. Bricsys is working hard to ensure that nearly all of the features in the Windows version operate properly in the Linux version.

Autodesk has a version of AutoCAD for Mac computers, but it has only about 85% of the commands found in the Windows version. Bricsys now ships their Mac version just after the Windows version comes out.

Here are the comparison charts from each CAD vendor for the functions included with the various operating systems:

AutoCAD Windows vs Mac: http://www.autodesk.com/products/autocad/compare/compare-platforms

BricsCAD Windows and Mac vs Linux: http://www.bricsys.com/en_INTL/bricscad/comparison/

SOLVING THE PROBLEM OF PORTING SOFTWARE TO LINUX

The part of the CAD system that deals with geometric objects is not a problem in porting. *Porting* is the term used to describe the process of making a software program work correctly with another operating system. The problems lie behind the scenes, specifically in the areas of programming interfaces and user interface elements.

Even for a large, wealthy firm like Autodesk, porting CAD programs to other operating systems is a difficult undertaking, because most of today's CAD software is intimately intertwined with the Windows operating system. Microsoft deliberately made it easy for programmers to write software for Windows, but then came the cost of making it excruciatingly difficult to tear away from Windows. For instance, a programming team at Autodesk took 18 months to rewrite AutoCAD for Mac and MacOS, and even then something like 30% of commands were left out of the initial release, as were most programming interfaces for third-party programmers.

Admittedly, a mere five years ago, no CAD programmer would have dreamed of writing code for anything other than Windows. Or perhaps for MacOS. (A few CAD firms, such as Graphisoft and Vectorworks, began on the Mac some twenty years ago, and since then developed their software simultaneously for MacOS and Windows. This foresight means no pain for them today!) Now, however, the plausible choices have quadrupled to include Android and iOS on portable devices, and Linux and MacOS on desktop systems — in addition to Windows on desktop and portable devices.

User Interface

To fix the two problem areas, Bricsys undertook two significant programming projects. The first rewrote the user interface using wxWidgets (http://www.wxwidgets.org). This interface allows BricsCAD to look the same on Linux, MacOS, Windows, and mobile operating systems.

"How should a ported program look?" This serious question faces software companies: should a CAD program look the same on all operating systems? If so, then current users feel comfortable switching. This is the approach Bricsys took, and so the Linux version looks the same as the Windows version.

Or should the CAD program look like the host operating system? If so, then new users feel comfortable starting with it. This is the approach Autodesk took with AutoCAD for Mac, which looks like a program written for MacOS, very different from the Windows version.

APIs

The second project was even more difficult, mimicking the Windows programming interface, something that no other CAD vendor attempted. (In the general computing world, there have been efforts like those of Wine, VMware, and Win4Lin to help Windows programs run on Linux and MacOS.) Programmers at Bricsys had to write the code for Linux that Microsoft normally provides for Windows.

Note that this problem affects only the parts of programming languages that depend greatly on the underlaying operating system, such as Visual LISP, .Net, and ARX or BRX. The OS problem does not affect customization internal to the CAD system, such as menu and toolbar macros, LISP routines, and scripts.

The end result ensures that add-ons written in Windows and Mac work in Linux . Here is a list of the APIs that Bricsys ported to BricsCAD for Linux:

All **LISP** functions, excluding VL, VLA, VLAX, and VLR functions, because they depend on Windows-only COM

All **DCL** functions

All **DIESEL** functions

All **TX** functions

All **BRX** functions, excluding interfaces that are strongly tied to Windows, such as AcUi/AdUi and OPM categories

All **SDS** functions, excluding Windows-specific types

TIP The **RecScript** command (script recorder) in BricsCAD produces *.scr* files that can be edited, which makes it more useful than the Action Recorder in AutoCAD. Since the Action Recorder's "scripts" cannot be edited, it is not really an API.

BENEFITS OF LINUX

Running the Linux operating system on computers instead of Windows has several benefits. These include the following items.

Linux is Free

Linux is free, as are subsequent upgrades. While Windows is included "free" with every new computer (actually, you pay a hidden cost of about $20), upgrades are not free. Upgrading from older versions of Windows can costs $40 to $200 per computer, depending on current offers available. (As this book is written, rumors abound that Microsoft may make Windows 10 upgrades free, as it did for Windows 8.1.)

Here is the annualized cost to upgrade OS licenses on 100 computers every three years, using the $70 upgrade price to Windows 8:

Windows OS	Linux OS	Savings
$7,000 every third year $2,333/year	$0 every year $0/year	$7,000 per 3 years $2,333/year

(Windows 10 upgrades are free for only the first year following its release; new installs are not free.)

Desktop Linux is now similar enough to regular Windows that some users cannot tell the difference. This is particularly true for those users who don't care about the UX (user experience), but instead care primarily about getting the work done. Once inside BricsCAD, the Linux version looks almost identical to the Windows version. Indeed, CAD operators at one design firm subsequently asked the IT staff to install Linux on their home computers, after experiencing its benefits at work.

Linux is Hardware-Efficient

Linux runs more efficiently than Windows. This means it can run CAD software faster on older hardware for more years than does Windows. Whereas Windows today can barely function on computers with "just" 1GB RAM, Linux has no problem with small amounts of memory. This is because Microsoft programmers were instructed by founder Bill Gates to assume computers have infinite memory and CPU speeds, which they do not. As a result, Windows was written inefficiently.

In contrast, Linux is based on Unix, an operating system from the 1970s, which was written with ultra-efficiency to run well on computers with very little memory and very slow CPUs. The ethos of efficiency has carried successfully into our current decade.

Linux Is Malware-free

Linux has fewer irritants than Windows and Mac MacOS. It does not suffer from malware attacks, such as viruses, since the number of Linux computers is too small for virus writers to bother with.

My favorite feature about Linux is that after updates are applied to Linux, I do not need to reboot the computer as I do with Windows or MacOS; I keep right on working. Even though Apple based MacOS on Unix, I am surprised that MacOS needs reboots following updates.

Here's a funny thing I have noticed: it is easier to get used to MacOS when you are already familiar with Linux, than coming directly from Windows.

Linux is Hardware-compatible

Linux runs on the same computers as Windows, unlike MacOS, which is locked to Apple hardware. To try out Linux, you can install it on an existing Windows computer; to try out MacOS, you have to buy all new hardware, and get used to different keyboard and trackpad interactions.

(A tip: If you have to get MacOS, save some money by buying the Mac mini with the maximum amount of RAM available, and then hook up your own monitor, keyboard, and mouse. I find the mini is more flexible than the MacBook.)

Linux Dual-boots

Linux has dual-booting built-in, unlike Windows. This means that one computer can run both Linux or Windows, through not simultaneously. When the computer starts, a Linux utility called "grub" lets you choose between running Linux or Windows. (MacOS also includes a dual-boot facility, called BootCamp for running Linux or Windows.)

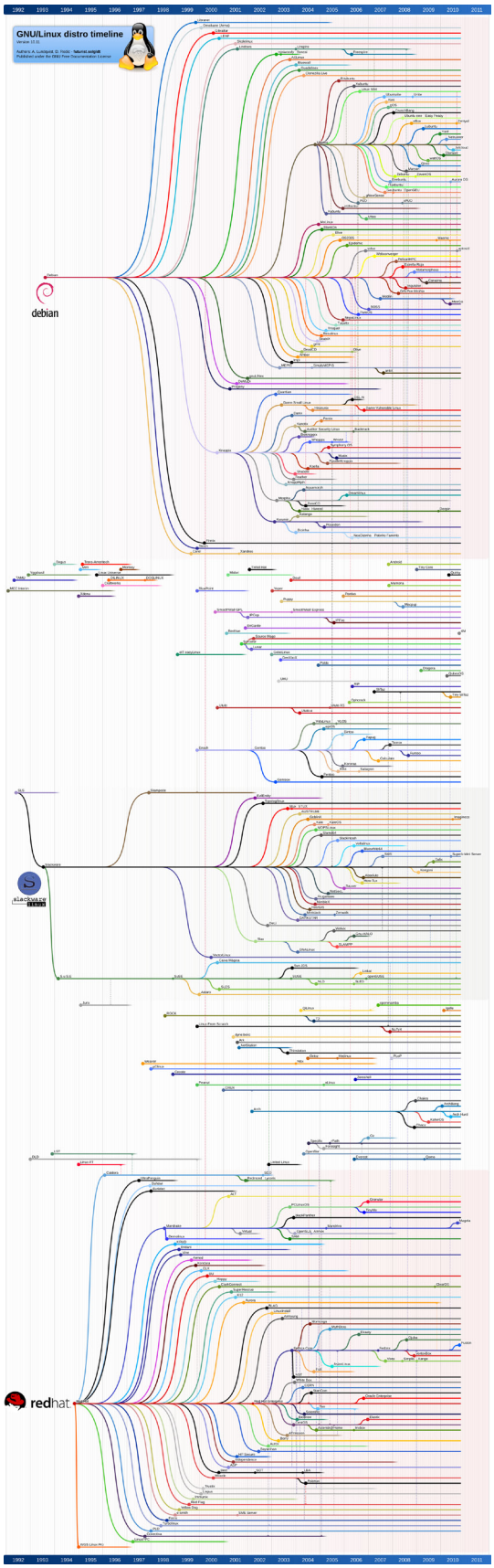
All my notebook computers are dual-booting; I usually run Linux, because it is more efficient. But when I need to use a program available only on Windows, then I shut down the computer and start it with Windows. The drawback to dual-boot is that it runs just one operating system at a time.

If you wish to run two (or more) at the same time, then you can use a free virtual manager program, such as Oracle Virtual Box (http://www.oracle.com/technetwork/server-storage/virtualbox/downloads/index.html#vbox). This program lets you run, say, Linux in a window (or full screen) inside Windows or MacOS, and even copy and paste between them.

DRAWBACKS TO LINUX

Linux never conquered the desktop the way it took over in all other areas of computing, such as Web servers, mainframe computers, smartphones, and embedded computing. Microsoft's monopolistic practices for many years were effective in locking out competitors, such as Apple and Linux.

Linux is confusing, because it can feel different from Windows, it has hundreds of versions and several graphical user interfaces from which to choose, and can sometimes have problems installing software.



Because it is different, it does not always have all the same software that Windows users are used to. Because there is so much choice in the number of versions of Linux, users can end up making no choice. And when software won't install, you won't use it.

Lack of Identical Software

Much of the basic software you run on Windows is available on Linux, such as Libre Office, which runs identically on Linux, MacOS, and Windows. If you use Microsoft Office on Windows, then you'll be running Libre Office on Linux. Other basics are also available in multi-OS versions, such as Web browsers (Chrome, Firefox, and Opera), image processing (Picasa), music and video playback (VLC), and Skype.

Linux comes with a ton of utilities; after all, it was written by geeks for themselves. For instance, the built-in screen grab software is much more sophisticated than the one for Windows or MacOS.

But it cannot run AutoCAD and other powerhouse software found in Windows and MacOS, such as PhotoShop and InDesign -- except through a Windows emulator, such as Wine. I find that emulators are not efficient (runs the software slower), are not 100% compatible (some software and some software functions don't operate), and development is patchy.

I recommend using native software, and I would rather do without than run software in an emulator or in a virtual machine. In this case, BricsCAD for Linux becomes the obvious choice.

Which Linux?

There are many more versions of Linux than there are of Windows. There is the source version written by Linius Torvald, after whom Linux is name. Then there are primary distributions, with names like Debian, Ubuntu, Gentoo, Fedora, Red Hat, Mandriva, and Slackware. See figure at left.

For every primary distribution, there are dozens of variants. This page at Wikipedia lists the names of more than 100 distributions and variants: http://en.wikipedia.org/wiki/List_of_Linux_distributions.

So, it can be hard — no, confusing — to choose one. (Here is a list of downloadable LiveCDs that contain Linux: en.wikipedia.org/wiki/List_of_live_CDs). In one way, it does not matter, since they all operate pretty much in roughly the same way; indeed, they work similarly to Windows and even more similarly to MacOS.

Because they are free, you can download a bunch of them and try them out. Downloads are often available as LiveCD format. You download the file (in *.iso* format), which you burn to a CD, and then you can run Linux from the CD drive and/or install onto a computer. In this case, I recommend using a virtual machine (VmWare or Virtual Box) to install a Linux distribution temporarily, unless you have a computer whose hard drive you can wipe. (You can run Linux off a CD or USB stick, but then it runs slowly, and you get a bad first impression!)

As for me, I use Mint Linux. It is based on the most popular dialect of Linux, Ubuntu, and so it can use *.deb* (Debian) installation files designed for Ubuntu. Better than just Ubuntu, however, Mint includes all the extras that make starting out with Linux less painful, such as common applications, drivers, and codecs. Download it free from https://www.linuxmint.com. For a version that runs in VirtualBox, see http://www.osboxes.org/linux-mint.

Problematic Installers

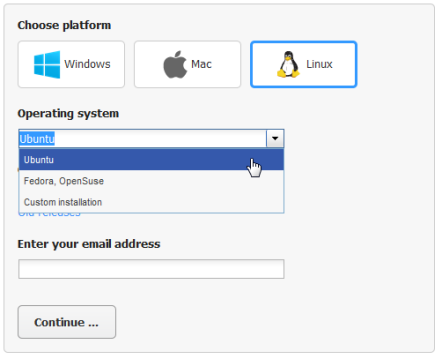
The biggest headache for new and medium-term Linux users is installing software. Many times, installing software goes without a hitch; other times, it does not work well and is a major pain. The problem exists because Linux first expected users to install software through the command-line interface; later, a GUI was added, and then different distributions came up with different ways of making installs easier. When you have hundreds of versions of Linux, you're bound to end up with dozens of installers. Someone once said in another context that more choice leads to less stress, but I disagree.

Major Linux vendors and software providers are fixing the problem in two ways: (a) through Windows-like installers, which operate nearly automatically; and (b) through MacOS-like software libraries built into the operating system.

Here is a list of the major distributions and the installer software they use:

Linux Distribution	Package File	Package Manager
Debian GNU/Linux	.deb	dpkg
Fedora Linux	.rpm	RPM
OpenSUSE Linux	.rpm	RPM
Others	.tgz	tar

At its Web site, Bricsys lists the download files in the following formats:



Choosing a BricsCAD for Linux variant to download

If the variant you used is based on Debian, then you click the DEB file button. I use Mint Linux, which is based on Ubuntu, which is based on Debian, and so I download *.deb* files.

Competing GUIs

If you have hundreds of dialects of Linux, then you are going to have several user interfaces. That's right: Linux offers easily replaceable graphical user interfaces. (This is also possible in Windows, but few have any desire to change Microsoft's design.)

There used to be a big split over which interface to use with Linux: KDE or Gnome. (I prefer Gnome.) Today, there is also Unity, which is designed for the smaller screens of netbooks and portable devices.

— — —

This chapter provided you with practical advice on running a design firm with both AutoCAD and BricsCAD, along with the pros and cons of replacing Windows with the Linux operating system.

CHAPTER SIX

Working in 3D

BRICSCAD PLATINUM IS PRICED LESS THAN AUTOCAD LT, YET IT PERFORMS 3D MODELING functions not found even in full-priced AutoCAD. How is this possible? Here are some reasons:

- Autodesk has high operating expenses; it must generate ever larger income for its shareholders
- Bricsys arranges its affairs to be a lean corporation
- AutoCAD must not compete against other, more profitable Autodesk software, like Inventor and Revit
- BricsCAD does not have to compete against other Bricsys software

The result is that BricsCAD over time gains more functions even as Autodesk pulls back on developing AutoCAD. Recent releases of AutoCAD were low in new functions, while the new features list for recent releases of BricsCAD go on for over a dozen pages. This chapter describes the many 3D capabilities of BricsCAD, and how they compare with AutoCAD:

- Direct modeling (press - pull)
- Quad cursor * , Tips widget *, and 3D mouse
- 3D geometric constraints * and dimensional constraints*
- Design intent *
- Mechanical browser, materials, and hardware library*
- Surface modeling
- Assembly modeling *
- Kinematic analysis *
- Sections
- Generative drafting (model documentation)
- Bills of material (data extraction)
- 3D compare *

* Functions missing from AutoCAD
Function shown in **blue** are new in BricsCAD V17

See Chapter 7 for information about the 3D add-ons modules for BIM, sheet metal, and Communicator translation.

3D FUNCTION COMPARISON

The table shows the BricsCAD edition in which 3D functions are found, and whether AutoCAD has similar functions. Those new since the last edition of this book are shown by blue dots.

	BricsCAD for Windows			BricsCAD for Mac & Linux			AutoCAD	LT
	Platinum	Pro	Classic	Platinum	Pro	Classic		
3D Compare	•			•				
3D Geometric constraints	•			•				
3D Surfaces	•	•	•	•	•	•	•	
3D Mesh modeling							•	
ACIS modeling and editing	•	•		•	•		•	(1)
ACIS viewing	•	•	•	•	•	•	•	•
Assembly modeling and editing	•			•				
Assembly viewing	•	•		•	•			
Bills of material	•			•			•	
Deformation modeling	•			•			•	
Design intent	•			•				
Direct modeling	•	•		•	•		•	
Generative drafting	•	•					•	
Hardware library	•	•						
Kinematic analysis	•			•				
Mechanical browser	•			•				
Section planes	•	•		•	•		•	
Surface modeling and lofts	•			•			•	
BIM modeling	Add-on			Add-on				
Import-export MCAD files	Add-on	Add-on		Add-on			•	
Sheet metal design	Add-on			Add-on				
Rendering	•	•		•	•		•	
Visual styles	•	•	(1)	•	•	(1)	•	
Walkthrough navigation	•	•	•				•	
3D mouse	•	•	•				•	•
Tips (Ctrl function) widget	•	•	•					

(1) Limited in function

TIP If some commands don't work, there are two reasons possible: you are running a lower edition of BricsCAD, such as Pro or Classic; only the Platinum edition has all commands described in this chapter.
If you are running BricsCAD Platinum and still cannot access some commands, it could be that the value of the **RunAsLevel** variable was changed. (The purpose of this variable is to simulate lower editions of BricsCAD by blocking higher level commands.) Enter **runaslevel**, then change the value to **2**.

BricsCAD's Direct Modeling vs AutoCAD's PressPull

BricsCAD can open 3D models made in AutoCAD, and then edit them. BricsCAD stores everything in a single .dwg file. BricsCAD uses the ACIS modeler licensed from Spatial; AutoCAD uses ShapeManager, an offshoot of ACIS. BricsCAD provides this set direct modeling commands.

Those commands new to V17 are shown in blue.

BricsCAD Direct Editing Operations	AutoCAD
dmChamfer chamfers edges	Chamfer
bimConnect creates L-connections between two solids	...
Copy copies parts and sub-entities	Copy
dmDelete erases parts and sub-entities	Erase
dmExtrude extrudes planar entities and sub-entities	Extrude
dmFillet rounds edges	Fillet
dmLoft creates lofts from curves	Loft
dmPushpull pushes and pulls faces and closed contours	PressPull
dmRevolve revolves planar entities and sub-entities	...
dmRigidSet3D turns components into a rigid set, like a group	...
dmTwist twists 3D objects along an axis	...
Boolean Operations	
Subtract subtracts one ACIS solid from another	Subtract
Union joins one ACIS solid with another	Union
...	Intersection
Kinematic Operations	
dmMove moves parts and sub-entities	...
dmRotate rotates entities and sub-entities	...
Modeling Assistance	
3dCompare compares differences between two models	...
dmDistance3d measures between the nearest points on boundaries, central points, or the axes of geometry on cylinders, circles, and spheres	
dmSelectEdges places faces and solids in a selection set	...
Ucs locates the UCS icon on entities	DUcs
dmUpdate updates 3D models to satisfy constraints	...
Help searches for help topics at the command line	Help

WORKING WITH DIRECT MODELING

Direct 3D modeling is the kind of design with which AutoCAD users are most familiar. It has been part of the venerable CAD program ever since 3D solid modeling was introduced to Release 13 in 1994. "Direct modeling" creates and edits 3D objects with no thought of their *history*. "History" is a record of the order in which the parts are made and edited, and the commands with which the 3D models are constructed.

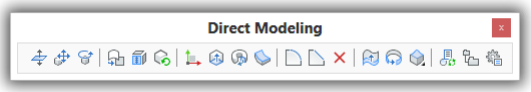
History-based modeling is the norm in MCAD packages like Inventor (from Autodesk) and Solidworks (from Dassault Systemes). The granddaddy of them all is Pro/Engineer (from PTC) being the first to popularize history-based parametric modeling in the late 1980s. While history-based modeling has proven to be beneficial in keeping track of the designer's intentions, the drawback is that large models become unwieldy to edit and can even crash; large models become painfully slow to edit as the history tree is updated with every change.

As computers became faster, however, CAD firms were able to implement direct modeling in a more powerful manner, and so it was re-popularized through a new breed of programs, like SpaceClaim and IronCAD. Old software firms like Autodesk and PTC also released new direct modeling software, with New Age names like Fusion and Creo, respectively.

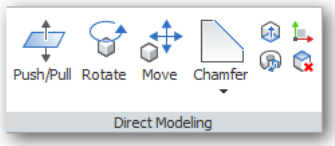
Bricsys rides this wave made possible by new algorithms, and so direct modeling is available in Pro and Platinum editions of BricsCAD, along with design intent and parametrics — everything, but the history tree. The CAD system works with all ACIS solids, including those imported from other MCAD systems.

Accessing Direct Modeling Commands

- Enter commands that start with 'dm'.
- Open the **Direct Modeling** toolbar and then chose a command



- In the ribbon's **Modeling** tab, look for commands in the **Direct Modeling** panel.

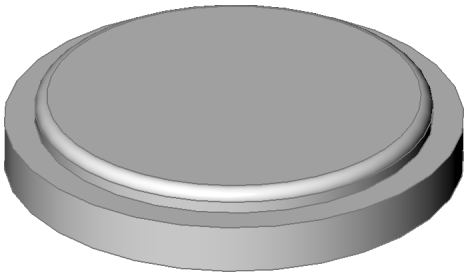


- In the **Model** menu, choose the **Direct Modeling** submenu

TIP The **dmStitch** command can be used to covert regions to surface objects.

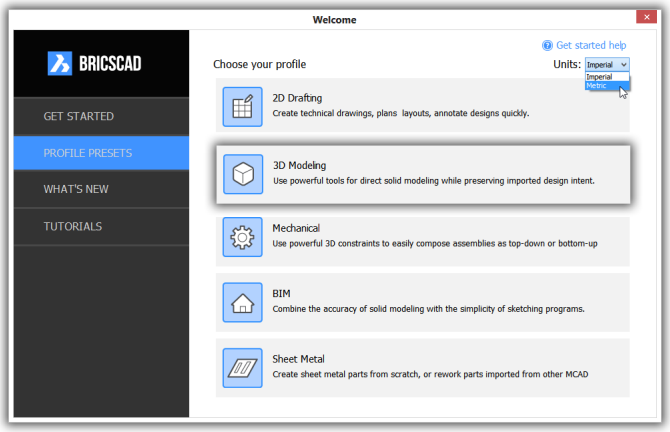
Direct Modeling Tutorial

To see how direct modeling works in BricsCAD, you'll design a lid for a storage container. The lid is 75mm round and 16mm tall. The smaller stopper portion is 65mm round x 8mm tall, and has a fillet.



Finished 3D model of a lid

1. Start BricsCAD with a new 3D Modeling drawing with metric units.

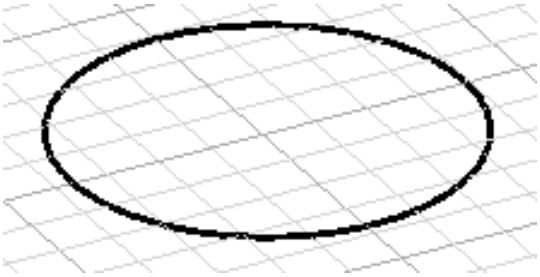


Starting a new drawing in 3D modeling workspace with metric dimensions

(If you are already in BricsCAD, then switch to **3D Modeling** workspace: right-click the workspace name on the status bar, and then choose "3D Modeling" from the shortcut menu.)

2. Draw the base of the lid as a circle 75mm in diameter, as follows:

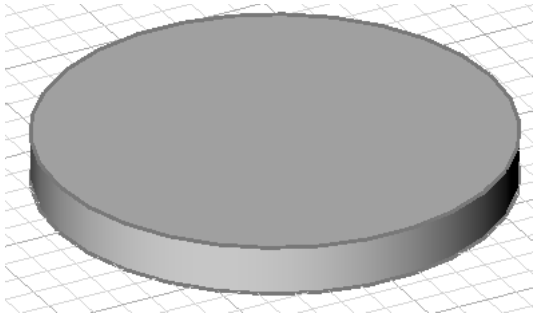
```
: circle
2Point/3Point/TanTanRad/Arc/Multiple/<Center of circle>: (Pick a point in the drawing)
Diameter/<Radius>: d
Diameter of circle: 75
```



Beginning with a circle

3. You extrud objects a couple of times in this tutorial, and so I'll show you different ways to do this. For the first extrusion, you use the official **dmExtrude** command. To extrude the circle into a cylinder 16m tall, start the command like this:

```
: dmExtrude
Select entities/subentities to extrude or set [MOde]: (Select the circle)
Entities/subentities in set: 1
Select entities/subentities to extrude or set [MOde]: (Press Enter to continue)
Specify height of extrusion or set [Auto/Create/Subtract/Unite/Taper angle/set Direction/set
Limit] <Auto>: 16
```



Extruding the circle to 16mm tall

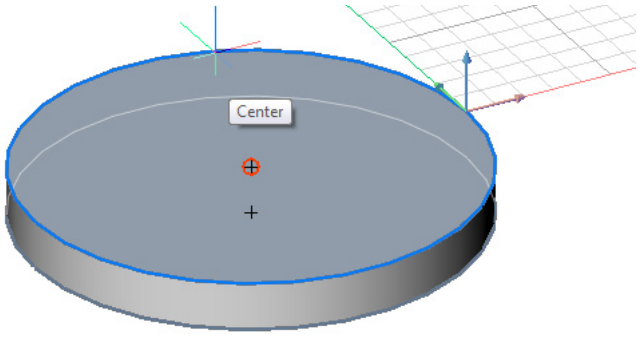
The next bit is to add a stopper to the lid. This is done in two steps: first, you draw a circle on top of the cylinder, and then you pull it up, creating the stopper in 3D. This time you'll use the Quad to extrude the circle into a cylinder.

4. First, through, draw the circle.

   a. Start the **Circle** command, and then enter CENter entity snap mode to locate the new circle at the precise center of the first curve:

```
: circle
Select center of circle or [2 Point/3 Point/TangenT-tangent-Radius/turn Arc into circle/
Multiple circles]:cen
Snap to centerpoint of: (Move cursor, as described below)
```

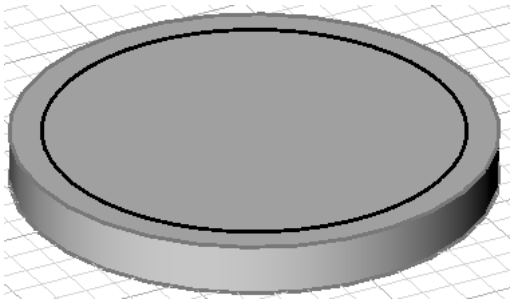
   b. Move the cursor to the top of the cylinder. Notice that it turns blue and that the grid jumps to the cylinder's top. This indicates that *dynamic UCS* is at work. (If BricsCAD doesn't do this, then click the **DUCS** button on the status bar to turn it on.) Dynamic UCS automatically relocates the 2D working plane in 3D space.



BricsCAD finding the center of the top of the cylinder

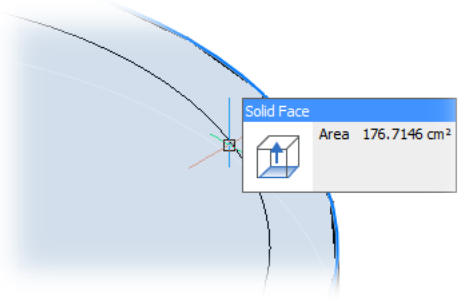
c. Draw a circle 65mm in diameter.

```
Set Radius or [Diameter] <75>: 65
```



Circle drawn on top of cylinder

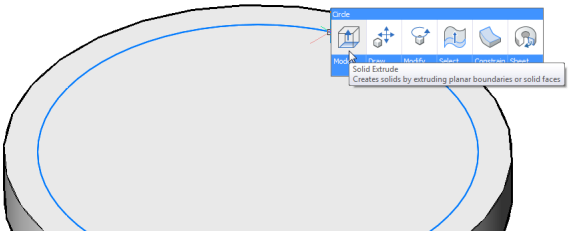
5. Move the cursor over the circle you just drew. Notice the Quad.



Quad appearing when cursor hovers over an entity

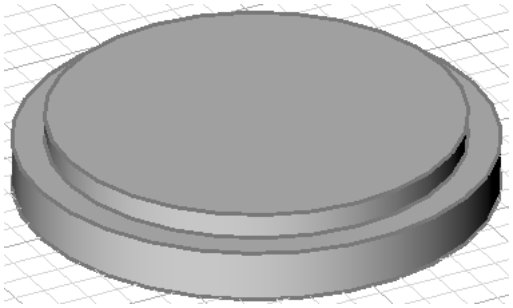
If the Quad does not appear, be sure to click the **QUAD** button on the status bar.

6. Move the cursor into the Quad. Notice that it expands to display a row of commands.



Moving the cursor into the Quad

7. Move the cursor over the icons until you find the command you need: Solid Extrude. Click the icon, and then enter **8** for the height, and then press **Enter**.

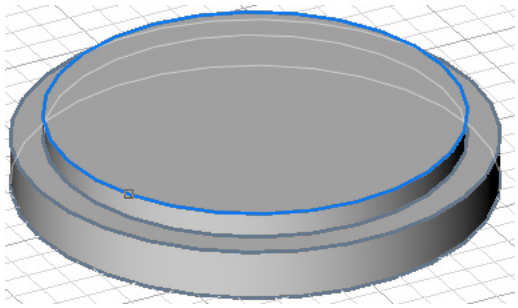


Second cylinder sitting atop the first

8. Round the edges with the **dmFillet** command, as follows:

   a. Enter the dmFillet command and then chose the edge to fillet:

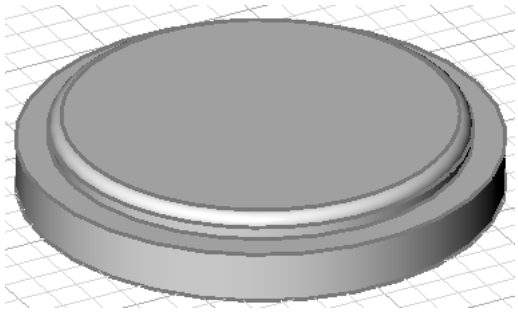
   ```
   : dmFillet
   Select edges to create fillet: (Select the edge highlighted by blue in the figure below)
   Entities/subentities in set: 1
   Select edges to create fillet: (Press Enter to continue)
   ```



Choosing the edge to fillet

   9. Specify a fillet radius of **4**.

   ```
   Specify fillet radius: 4
   ```



Completed lid with filleted edge

6. To view the lid from a variety of angles dynamically, hold down the **Shift** key and then move the mouse while holding down the center button (or roller wheel) — just as in AutoCAD.

Workspaces, 3D Viewing, Quad Cursor, & 3D Mouse

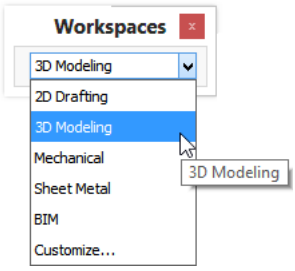
BricsCAD provides many ways to view models in 3D. I describe some of them in this section.

To switch between 2D and 3D drafting environments, BricsCAD uses the same concept of "workspaces" as AutoCAD. BricsCAD comes with the following workspaces; the table compares equivalent workspace names between BricsCAD and AutoCAD:

BricsCAD Workspace Names	Equivalent AutoCAD Workspace Names
2D Drafting	Drafting and Annotation
3D Modeling	3D Modeling
...	3D Basics
Mechanical	...
BIM	...
Sheet Metal	...

Accessing the Workspace Commands

- Enter the **WsCurrent** command
- Open the **Workspaces** toolbar and then chose a workspace



- Right-click the current workspace name on the status bar, and then choose another one

VIEW ROTATION & UCS FACE COMMANDS

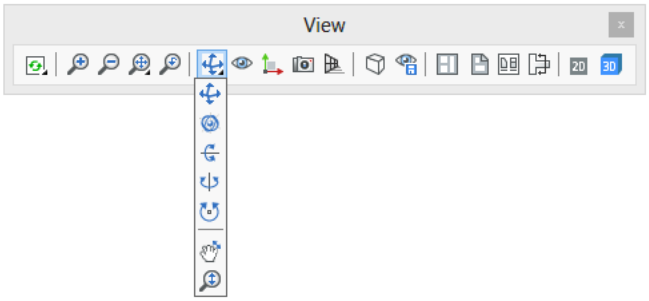
Modeling in 3D is just like drawing in 2D: we still work on a 2D plane for the most part. The 2D plane often is the face of a 3D object. Because 3D objects typically have six or more faces, it is important to move to the correct face quickly. For enable this, BricsCAD has dynamic UCS, which like AutoCAD forces the UCS onto the selected plane. To turn on this function, click the **DUCS** button on the status bar.

In addition, BricsCAD has view rotation commands to swivel our view around the 3D model. Some are the same as in AutoCAD but have different names, as the table below indicates:

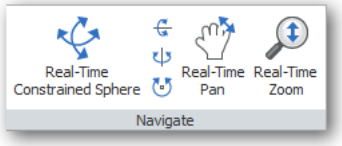
BrisCAD	AutoCAD	Description
RtRot	3DOrbit	Rotates the 3D view dynamically
RtRotCtr	3DCOrbit	Rotates the 3D view about a user-defined center point
RtRotF	3DFOrbit	Rotates the 3D view freely
RtRotX	...	Rotates the 3D view about the screen's x-axis
RtRotY	...	Rotates the 3D view about the screen's y-axis
RtRotZ	...	Rotates the 3D view about the screen's z-axis

Accessing the 3D Viewing Commands

- Enter the commands listed in the table above
- Open the **View** toolbar, and then click the **Real Time** flyout



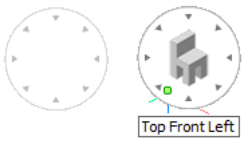
- In the ribbon's **View** tab, choose commands from the **Navigate** tab



- From the **View** menu, choose **Real Time Motion**

BRICSCAD'S LOOKFROM VS. AUTOCAD'S VIEWCUBE

AutoCAD has the navigation cube for quickly changing 3D viewpoints; in BricsCAD, it is known as the LookFrom widget. Moving the cursor into the widget's circle displays the preview of a chair; clicking the triangle changes the 3D viewpoint.

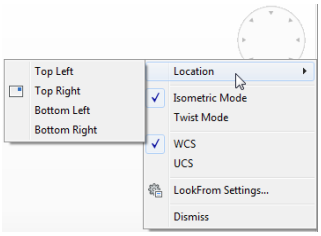


***Left:** Lookfrom widget at rest; **right:** with cursor entering the circle*

Click a triangle to change the viewpoint, such as Front Left or Right. Hold down the **Ctrl** key to access the bottom views. The green dot indicates the cursor position, kind of like a laser pointer.

Left: *LookFrom control in BricsCAD;* ***right:*** *equivalent ViewCube control in AutoCAD*

The easiest way to change how the LookFrom control operates is to right-click the control, and then choose an option from the shortcut menus:



Context menu for the LookFrom control

The LookFrom control operates in two modes, isometric and twist. The difference is how they rotate the 3D viewpoint:

- **Isometric** mode is like using the Viewpoint or View commands
- **Twist** mode is like using the RtRotF (3DOrbit in AutoCAD) command

Left: *Isometric mode;* ***right:*** *Twist mode*

TIP When in Twist mode, click the center of the LookFrom control to return the view to its home view.

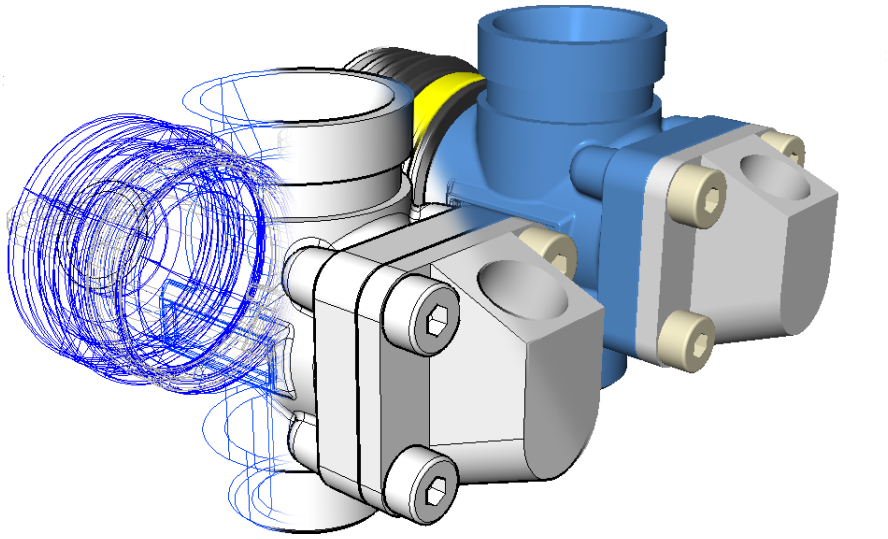
See Chapter 2 “Comparing User Interfaces” for more about the LookFrom widget.

Accessing LookFrom Commands

- Enter the **LookFrom** command or press the **Ctrl+Shift+L** keyboard shortcut
- From the **View** menu, choose **LookFrom**
- Right-click the LookFrom widget, and then choose an option from the shortcut menu

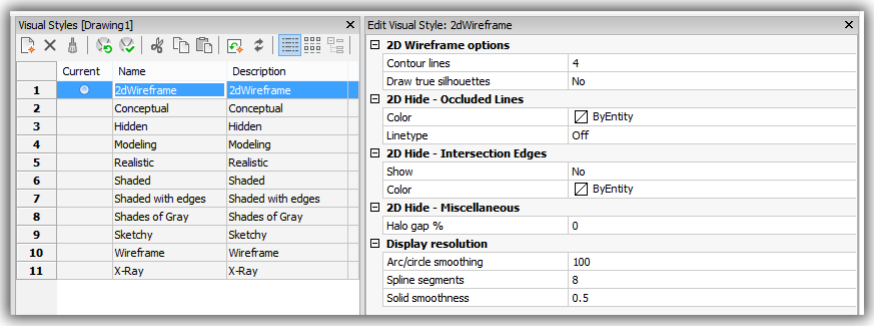
VISUAL STYLES AND RENDERING

Three-D modeling means that objects can be rendered to look lifelike — or even artificial. BricsCAD offers visual styles so that you can draw and edit in rendered mode. You customize styles through the Drawing Explorer. AutoCAD has the same system of customizable visual styles, but offers fewer presets styles.



Left to right: *Wireframe, shades of gray, and rendered visual styles*

See chapter 3 for a comparison table of named visual styles available in both CAD packages.



Parameters for visual styles

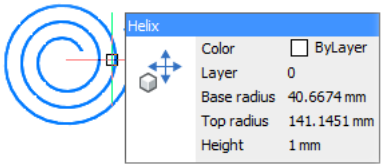
Accessing Visual Styles Commands

- Enter the **VisualStyles** command
- From the **View** menu, choose **Visual Styles**
- From the **Tools** menu, choose **Drawing Explorer**, and then Visual Styles

WORKING WITH THE QUAD CURSOR

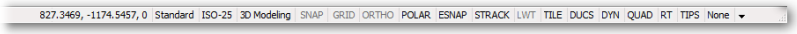
All editions of BricsCAD provide the Quad cursor. (AutoCAD has nothing similar.)It provides intuitive access to contextual commands. The Quad cursor changes its content, depending on the context. Contexts that affect the Quad cursor include drawing/editing and the workspace:

1. When you first "hover" of an object (entity), the Quad appears and lists some of the properties of the object. *Hover* means that the cursor is over an object, but the object is not picked with a click. You can change the properties displayed by the Quad through the Customize command; see chapter 4.



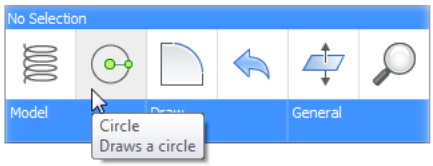
Entity is not selected, so Quad shows some of its properties

TIPS If you do not see the Quad, then click the **QUAD** button on the status bar.



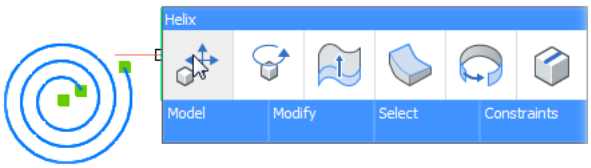
If you do not see properties in the Quad, click the **RT** button on the status bar; "RT" is short for rollover tooltips.

2. When no objects are selected, right-click to put the Quad cursor into drawing mode.



Drawing commands in the Quad when right-clicking an empty spot in the drawing

3. Selecting an object put the cursor into editing mode. The content of the cursor changes, depending on which on the entity and the current workspace.



Initial set of editing commands displayed the by Quad

4. To see more commands, move the cursor into one of the blue tabs, such as "Model" or "Modify."

TIPS When you right-click an empty area of the drawing with QUAD turned off, BricsCAD repeats the last command.
The first icon displayed by the Quad is the command that was last used.

Accessing the Quad Command

- Enter the **QuadDisplay** command
- Click **QUAD** on the status bar
- Drawing mode: right-click an empty part of the drawing
- Editing mode: pause the cursor over an entity, with no command running

TIPS BAR

The Tips bar is a BricsCAD user interface element that appears during at certain times, such as during certain 3D modeling operations. It allows you to select a command option without using the keyboard. (AutoCAD has nothing like this.)

The Ctrl icon reminds you to tap the **Ctrl** key to move through the options listed in the bar:

Left to right: *Tips bars displayed for several commands*

For example, the bar illustrated on the left appears with the **dmExtrude** command. The four icons are for the Auto, Create, Subtract, and Unite options. Clicking the **x** dismisses the bar; it does not cancel the command. You can use the **Ctrl** key to switch between options.

Toggle the display of the Tips bar with the **TIPS** button on the status bar.

BRICSCAD COMMAND PREFIXES

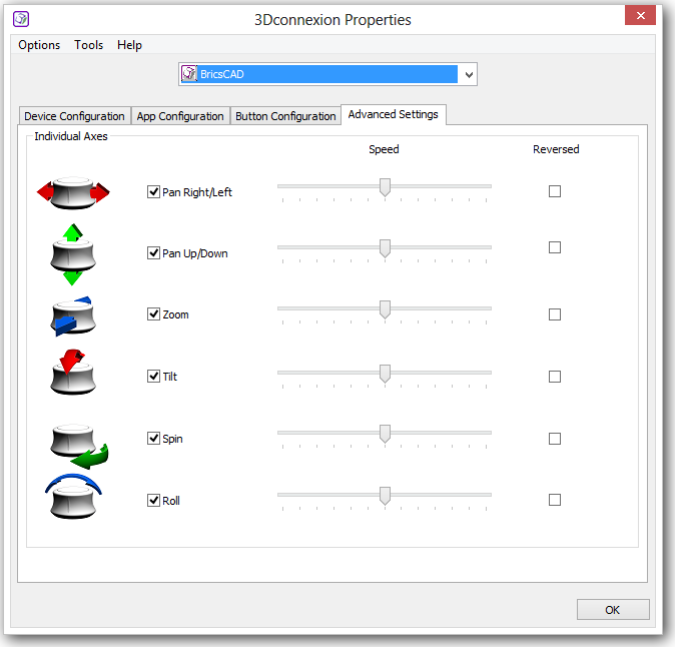
Bricsys uses a number of prefixes to identify the purpose of related commands:

bim	BIM (building information modeling) commands,such as bimClassify
bm	BricsCAD Modeling commands, such as bmInsert
dm	Direct Modeling commands, such as dmRepair
sm	Sheet Metal commands, such as smLoft

WORKING WITH A 3D MOUSE

BricsCAD supports a 3D mouse when it is plugged in and the 3dconnexion driver is installed and running. AutoCAD also supports 3D mice. While AutoCAD provides access to 3D mouse functions, BricsCAD does not; its sole option is the **Ctrl3DMouse** variable, which toggles use of the 3D mouse.

External to BricsCAD, use the 3Dconnexion Properties dialog box to set the movements of the mouse's puck and actions of the its buttons.



3Dconnexion control panel determines how BricsCAD reacts to the 3D mouse

To access this dialog box in Windows 7, click the **Start** button, and then choose **All Programs | 3Dconnexion | 3D Mouse Control Panel**, and then click **Properties**. In Windows 8, press **Windows** button and **Q** to access the Search field; search for "3dcon" and then choose the 3Dconnexion Control Panel app that appears in the results. In Windows 8.1 and 10, click the start button and then start typing "3dcon..." until the program appears in the search results.

3D Geometric & 3D Dimensional Constraints

Working with 3D constraints in BricsCAD is just like working with 2D constraints in AutoCAD. The difference is that they also operate in the z-direction. (AutoCAD has no 3D constraints.)Expressions and parameters can specify values and formulae for 3D dimensional constraints, just as AutoCAD does for 2D constraints.

The 3D constraints are available in the Pro and Platinum editions of BricsCAD. The difference is that while the Pro version can solve constraints, only the Platinum edition can apply them.

3D Dimensional Constraints
dmAngle3D applies 3D angle constraint
dmDistance3D applies 3D distance constraint
dmRadius3D applies 3D radial constraint

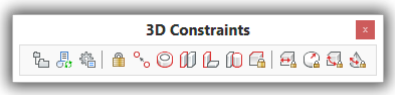
3D Geometric Constraints
dmCoincident3D applies 3D coincident constraint
dmConcentric3D applies 3D concentric constraint
dmConstraint3d is a super command that applies any kind of 3D constraint
dmTangent3D applies 3D tangency constraint
dmFix3d applies 3D fix constraint
dmParallel3D applies 3D parallel constraint
dmPerpendicular3D applies 3D perpendicular constraint
dmTangent3D applies 3D tangency constraint

WORKING WITH 3D CONSTRAINTS

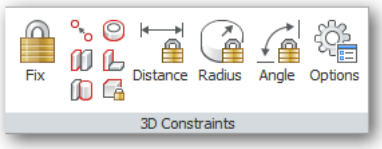
For a tutorial on using 3D constraints, see the Assembly Drawings section later in this chapter.

Accessing 3D Constraint Commands

- Enter the commands listed in the table above
- Open the **3D Constraints** toolbar



- In the ribbon's **Parametric** tab, select commands from the **3D Constraints** panel



- From the **Parametric** menu, choose **3D Constraints**

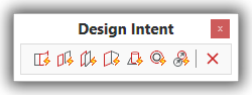
3D Design Intent

BricsCAD Platinum determines automatically what you were probably intending to design. This is known as *design intent*. When design intent is turned on, BricsCAD recognizes parts of 3D entities that ought to be edited together automatically. This is similar to the actions of another MCAD program known as Solid Edge, where the function is named "Live Rules." AutoCAD does not provide design intent.

Consider an object with several holes of the same size. When design intent is running, it recognizes that they all have the same diameter. When you change the diameter of one of the holes, BricsCAD changes the diameters of the others automatically. This is why design intent is also known as "automatic 3D geometry constraints recognition."

Unlike constraints, you cannot, unfortunately, apply design intent to specific areas of a model: design intent is universal. You can choose, however, which aspects of design intent you want operating. For instance, you can have BricsCAD recognize planes that are just parallel, coincident, or perpendicular to each other.

I find it convenient to toggle settings through the Design Intent toolbar. Click the big red X to turn off design intent.



Design Intent toolbar toggles settings

Design intent settings are toggled through the **dmRecognize** variable; see table below. Setting the value negative turns off design intent, but retains the former value.

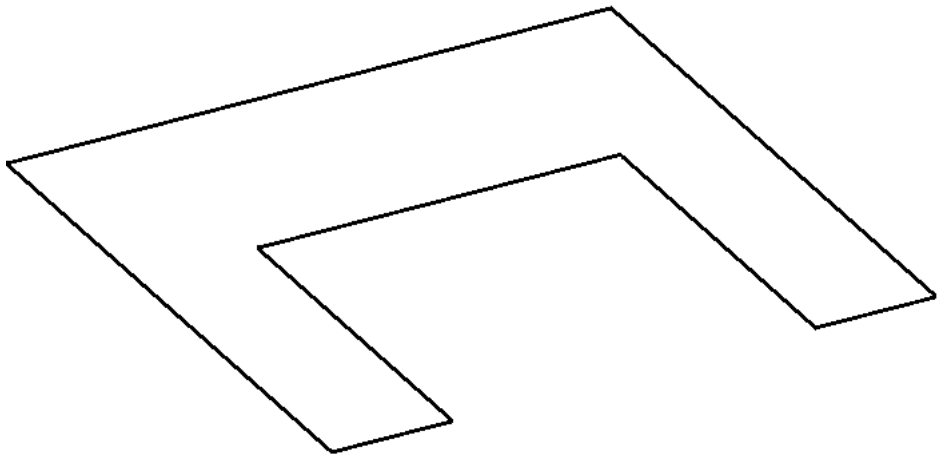
dmRecognize	Description	On by Default
0	All off	
1	Tangent surfaces of planes, cylinders, and cones	
2	Coincident planes	•
4	Parallel planes	•
8	Perpendicular planes	
16	Cylinders perpendicular to planes	
32	Coaxial surfaces of cylinders and cones	•
64	Equal radius on cylinders (or holes) and spheres	•
negative value	All off, yet retains value of the previous setting	

There is a limitation to automatic feature recognition that's common to all CAD systems: the engine works only with 3D solids that it recognizes. For BricsCAD, this means that design intent works with simple shapes — planes (flat faces), cylinders, cones, spheres — but not with bodies of arbitrary shape. The simple shapes can, however, be part of a more complex body.

WORKING WITH DESIGN INTENT

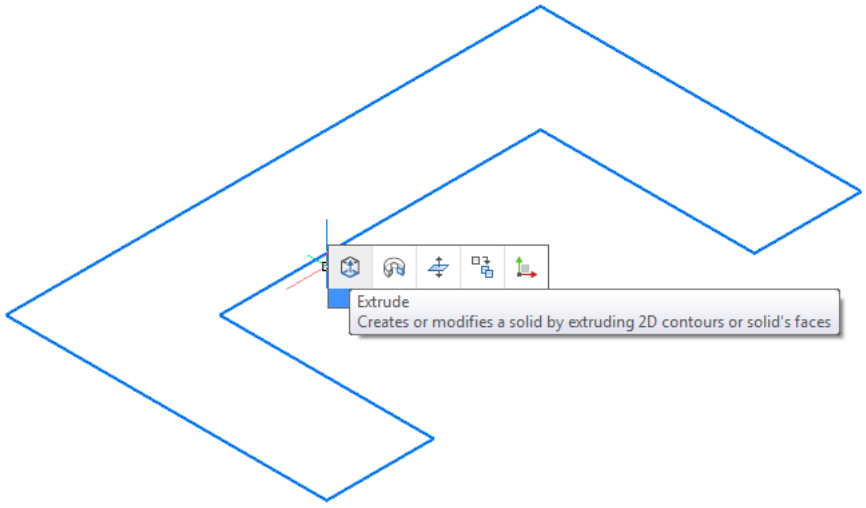
To show how design intent works in BricsCAD, you draw a 3D shape and then use the **dmPushPull** command without — and with — design intent turned on.

1. Start BricsCAD in "3D Modeling" workspace.
2. Draw a 2D shape with the **PLine** command, similar to the one shown below. The exact size does not matter for this tutorial.



Closed polyline drawn with the PLine command

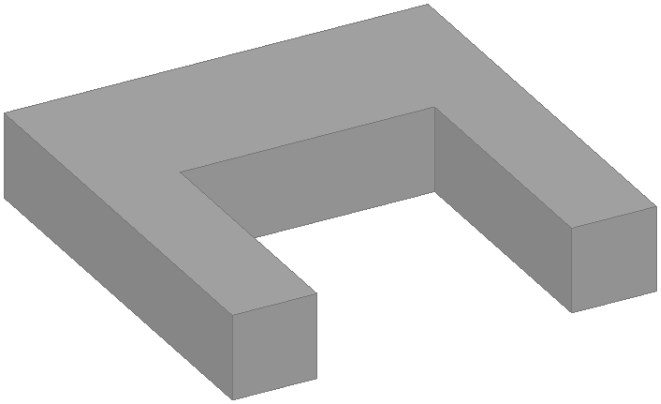
3. Turn the 2D shape into a 3D model by executing the **Extrude** command from the Quad cursor. To do so, follow these steps:
   a. Move the cursor over the polyline. Notice that it turns blue to indicate it is selected by default.
   b. Move the arrow cursor into the Quad cursor icon; notice that it expands.



Exposing the Quad cursor over the polyline

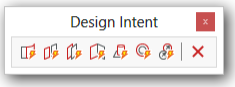
c. Click the **dmExtrude** button. (It is not necessary to select the polyline, a benefit to using the Quad cursor to execute commands.)

```
Specify height of extrusion or [Direction/Path/Taper angle] <1>: 10
```



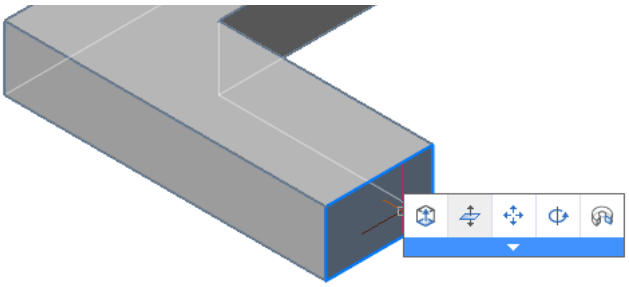
Polyline extruded into a 3D model with the Extrude command

4. Open the Design Intent toolbar:
   a. Right-click any toolbar, and then choose BRICSCAD | Design Intent.
   b. Ensure design intent is turned off by clicking the red X button at the end of the toolbar.



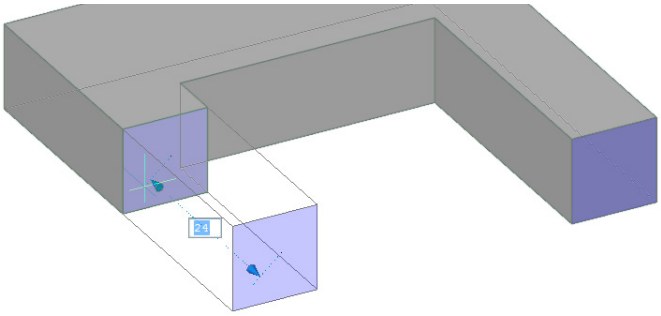
Click the last button on the right to turn off all design intent modes

5. Now you will change the length of one arm with design intent turned off. From the Quad cursor, access the **dmPushPull** command.



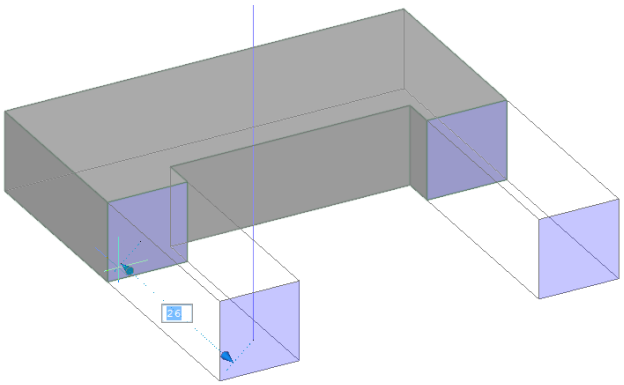
Choosing the dmPushPull command from the Quad cursor

6. Drag the face indicated by the figure below. Notice that the coincident face remains in place.



Dragging one face with the dmPushPull command

7. In the Design Intent toolbar, turn on **Coincident Planes**.
8. Repeat the **dmPushPull** command to see the effect of design intent on your editing operations. As you drag one face, notice that the coincident face moves along.

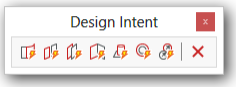


Both planes move together when Coincident Planes is turned on

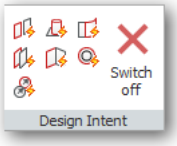
BricsCAD recognized that the other edge was in the same plane as the first one, and so moved it simultaneously and automatically. Should you wish this to not occur, simply turn off design intent.

Accessing Design Intent Commands

- Enter the **dmRecognize** variable
- Open the **Design Intent** toolbar



- In the ribbon's **Parametric** tab, look for commands in the **Design Intent** panel.

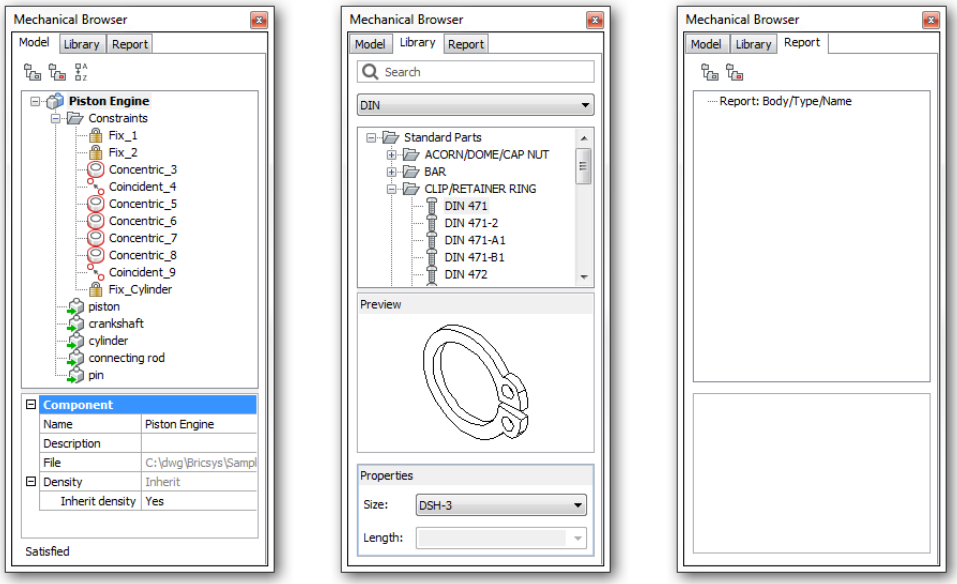


- Enter the **Settings** command and then go to the **Drawing | Drafting | Direct Modeling** section

Mechanical Browser & Hardware Library

BricsCAD Platinum offers the Mechanical Browser panel (palette) to perform many duties, as listed below. The nearest AutoCAD has to Mechanical Browser is the Parametric Manager palette for entering formulae.

- **Model** tab keeps track of parts in assemblies (not available in AutoCAD), lists the constraints that are attached to parts, and records formulae for dimensional constraints
- **Library** tab accesses a library of 30,000 mechanical parts in a variety of international standards (not available in AutoCAD)
- **Report** tab reports on problems found in models



Left: *Mechanical Browser bar open at the Model tab;* ***center:*** *Library tab;* ***right****: Report tab*

WORKING WITH THE MECHANICAL BROWSER

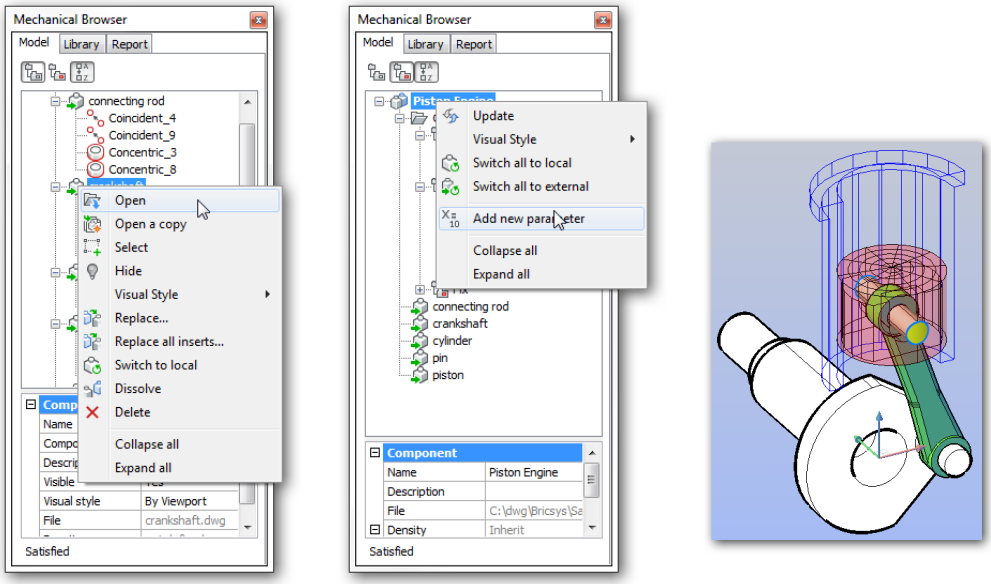
The Mechanical Browser comes into effect when working with 3D models, assemblies, and sheet metal designs. It lists the parts of the models and the constraints used to hold the parts together.

The toolbar displays the model tree in different ways:

Mechanical Browser's toolbar

Group by entity lists each entity in alphabetical order together with a set of constraints, if any
Group by type lists all constraints first, and then all entities in alphabetical order
Alphabetic sort the list in obverse and reverse alphabetical order

Right-click a node to access a shortcut menu that contains most of the commands available in the browser.

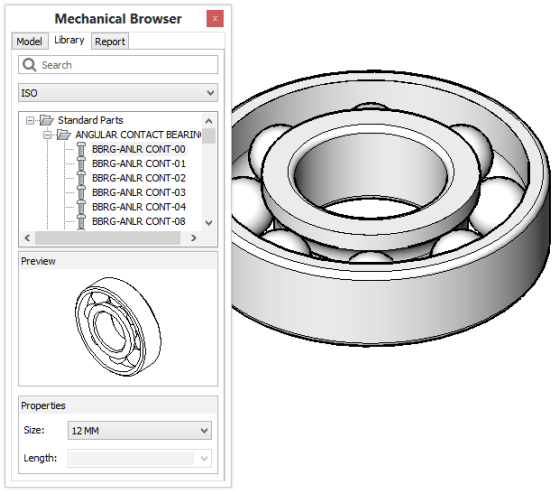


Left: *Accessing the context menu for parts;* ***center:*** *Accessing the context menu to add formulae;*
Right: *Assembly with each part shown in a different visual style*

BricsCAD V16 introduced local mechanical components, which are stored in the same *.dwg* file where the component is inserted in the mechanical assembly.

WORKING WITH THE HARDWARE LIBRARY

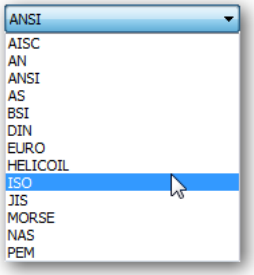
The Hardware Library provides 30,000 parts in parametric form. “Parametric” means that you specify the size of a selected part, and then BricsCAD generates it. AutoCAD does not include a parametric hardware library, but provides access to them online through its Seek command.



Part inserted in the drawing

To pick a part from the library, follow these steps:

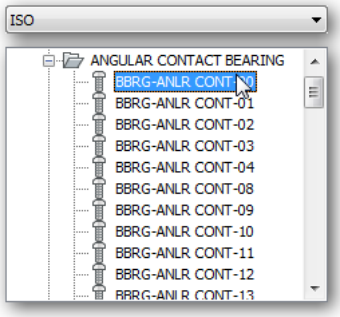
1. From the droplist, choose a standard, such as **ISO**. (The Search field searches only part names within the current standard; it does not search the entire library.)



Choosing an international standard for the parts

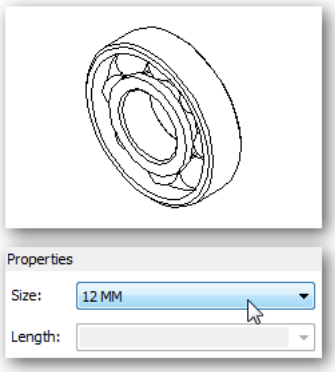
TIP The standard you choose determines the number of parts available. For instance, the JIS standard has the longest list at 47 part types, the ANSI standard has 28 part types, while AN has just one.

2. Chose a part type, such as **Angular Contact Bearing**, and then a specific model, such as **BBRG-ANLR CONT-00**. (Click **+** to open the node.)



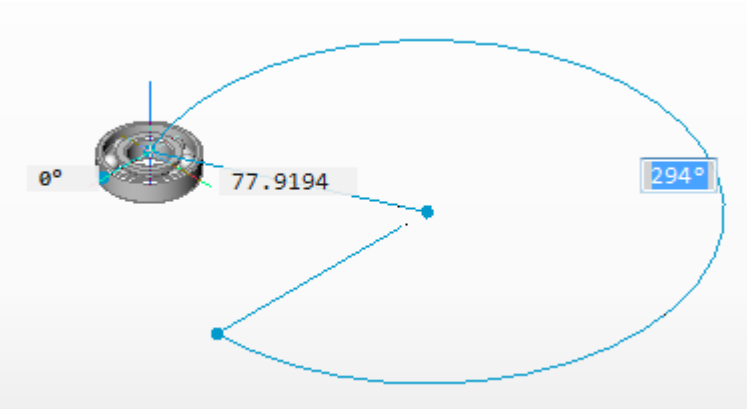
Choosing a part type

3. Notice the preview image that appears at the bottom of the bar, as well as the properties. The properties are the *parameters*, with which you specify the size of the part. (The properties available depend on the part selected; not all properties can be modified.) Accept the default properties by changing none of them.



Viewing the preview and specifying the parameters (Properties)

4. To place the component, drag it its *name* "BBRG-ANLR CONT-oo" into the drawing.



Place the component in the drawing

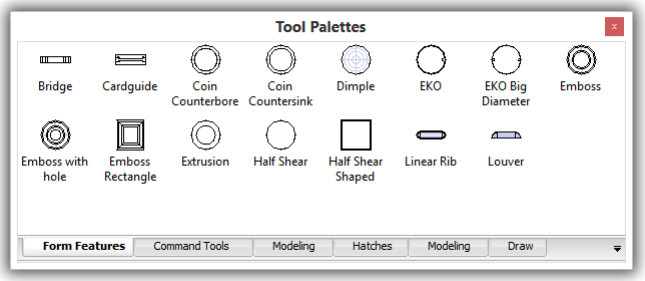
5. When you let go of the mouse button, prompts appear in the Command bar. You can ignore all but the last one:

```
Insertion point [Rotate/Base point/Name/mUltiple] <0,0,0>: (Specify a point, or enter an option)
```

Placement Options
Insertion point specifies the x,y,z coordinates of the part's location in the drawing
Rotate rotates the part about the insertion point
Base point changes the base point to another spot on the part
Name changes the name from the default one generated by the library; this is the name that appears in the bill of materials
mUltiple repeats the prompts to insert the part more than once

Tool Palettes

The Tool Palettes panel can also be used to access commonly-used parts for 3D modeling. To place the parts, drag them from the panel into the drawing. (NEW TO V17) The Form Features tab contains 15 3D parametric parts useful for sheet metal design.



Tool Palettes with form features

Accessing the Mechanical Browser

- Enter the **bmBrowser** command
- Right-click any toolbar or ribbon tab, and then choose **Mechanical Browser** from the shortcut menu

Modeling and Deforming 3D Surfaces

BricsCAD performs surface modeling using direct modeling commands and 3D constraints. These commands also work with 3D solids, but now produce or edit surfaces, depending on the context. AutoCAD also does surface modeling, although it lacks 3D constraints. These are true surfaces, which can be deformed, and are not the older mesh surfaces found in BricsCAD and AutoCAD since the 1980s, such as those made with the Ai_Box and PFace commands.

Creating and editing surfaces in BricsCAD works just like in AutoCAD. The vertices, edges, and faces of surfaces are deformed with the same commands used to deform 3D solids.

Use the following commands to create surfaces:

BricsCAD Surface Modeling Commands
dmDeformCurve deforms surfaces by moving or rotating edges to a specified set of target curves
dmDeformMove moves or rotates edges of surfaces
dmDeformPoint transforms points lying on specified faces
dmDelete removes holes (open loops) and faces from surfaces
dmExtrude extrudes curves, edges, planar entities, and faces into 3D surfaces
dmRevolve revolves curves, edges, planar entities, and faces into 3D surfaces
dmStitch stitches a set of surfaces into a single 3D surface
dmThicken convert surfaces to 3D solids with a specified thickness

TIP When extruding or rotating a 2D entity, BricsCAD converts them automatically depending on their type:
Open 2D entities become 3D *surfaces*
Closed 2D entities become 3D *solids*
To turn a 3D surface into a 3D solid, use the **dmThicken** command.

Accessing Surfacing Commands

- Enter then commands listed above
- From the ribbon's **Surfaces** tab, choose a command

3D Assembly Modeling

BricsCAD Platinum creates and edits assemblies. "Assemblies" are parts that stuck together using 3D constraints to create larger, more complex models. Indeed, assemblies are impossible without 3D constraints. This same thing happens in expensive programs Autodesk's Inventor or Dassault's Solidworks software. (AutoCAD cannot do this, while the Pro edition of BricsCAD is limited to displaying assemblies.)

An assembly is made from two or more parts that Bricsys calls "components." Components can be sourced from the following places:

- Regular DWG files converted to components through the **bmInsert** command
- Parts inserted from the Mechanical Browser's Hardware tab with the **bmHardware** command
- Parts drawn from scratch using BricsCAD's 2D and 3D modeling commands, then converted to components with the **bmForm** command

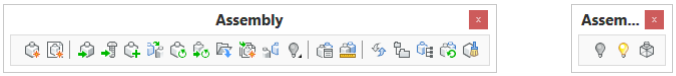
Assemblies can contain assemblies of components. Individual components can be hidden or shown. A nice touch is that each component can have its own visual style, meaning some can be see-through and some opaque.

Assembly Modeling Commands
bmDependencies lists names of files containing component definitions in the assembly
bmDissolve dissolves mechanical components inserted into drawings
bmExternalize converts local components to external components
bmForm forms a new mechanical component and inserts it into the drawing
bmHardware and **-bmHardware** insert standard hardware parts as mechanical components
bmHide hides mechanical components
bmInsert and **-bmInsert** insert existing mechanical components into drawings
bmLocalize converts external components to local components
bmMassProp calculates mass properties of components; takes into account densities
bmMech converts the current drawing into one suitable for assembly construction
bmNew creates a new mechanical component as a new drawing
bmOpen opens a part from an assembly for editing
bmOpenCopy creates new drawing with a copy of selected components
bmRecover recovers mechanical assemblies
bmReplace replaces component inserts
bmShow shows hidden mechanical components
bmUnmech converts mechanical components into plain drawings
bmUpdate updates the hierarchy of mechanical components
bmXConvert converts X-Hardware solids into mechanical components
bmVStyle specifies the visual style of individual components

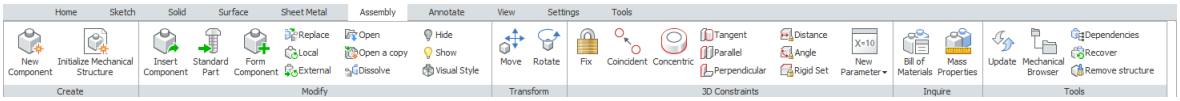
Other Commands
bmBom inserts a bill of materials (BOM) table into the drawing
bmBrowser opens and closes the Mechanical Browser bar

Accessing Assembly Commands

- Enter the commands listed in the table above
- Open the **Assembly** and **Assembly Visualization** toolbars



- In the ribbon's **Assembly** tab, choose a command

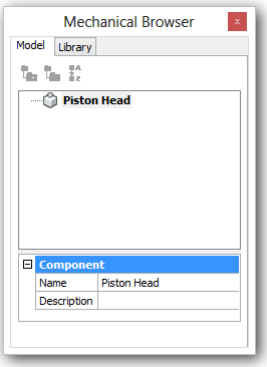


- From the **Assembly** menu, choose a command

WORKING WITH ASSEMBLIES

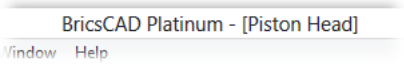
In this tutorial, you create a simple assembly of two parts: a pin and a piston. Step 3 is the crucial one, because that is where you turn a regular drawing into an assembly drawing.

1. Start BricsCAD in the **3D Modeling** workspace.
2. Open the Mechanical Browser bar with the **bmBrowser** command.
3. To turn the plain DWG drawing into an assembly drawing, follow these steps:
   a. In the Mechanical Browser, click the **Name** field (located near the bottom of the browser).
   b. Edit the text so that "Drawing1" reads **Piston Head**.



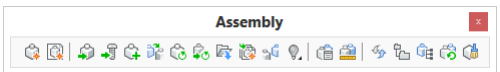
Drawing renamed by the Mechanical Browser

   c. Notice that BricsCAD changes the name of the drawing to match. Press **Ctrl+S** to save the drawing.



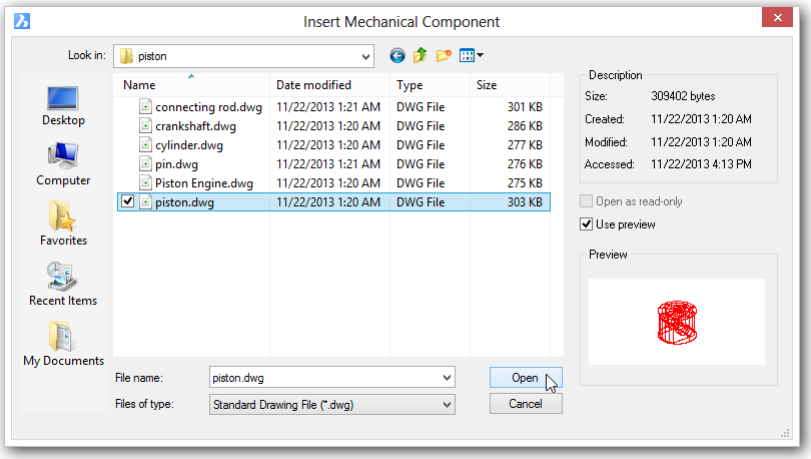
Drawing renamed in the title bar

4. With the drawing prepared for assemblies, the next steps are to insert a pre-drawn component into the drawing. Follow these steps:
   a. Open the Assembly toolbar by right-clicking a toolbar and then choosing **BRICSCAD | Assembly**.



Assembly toolbar for inserting components

b. Click **Insert Component**. Notice the Insert Component dialog box.

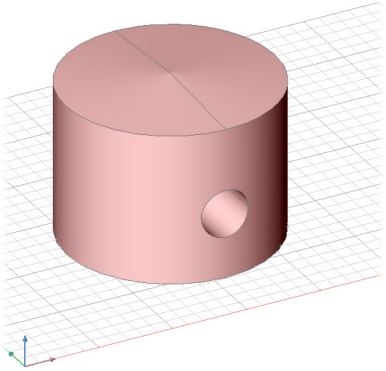


Choosing a DWG file to insert as a component of an assembly

c. Navigate to the *Samples* folder to access mechanical drawings provided with BricsCAD:
`C:\Program Files\Bricsys\BricsCAD V17\en_US\Samples\Mechanical\piston`

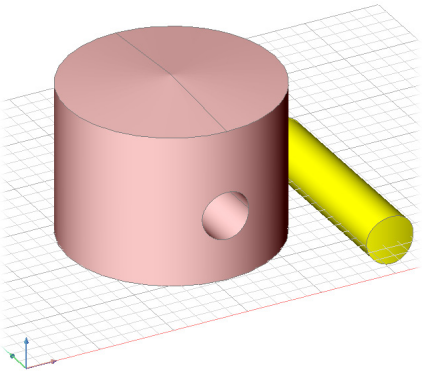
d. Select the *piston.dwg* file and then click **Open**.

e. Place the piston at any convenient spot in the drawing; the exact location is immaterial.



Piston placed as a component in the assembly drawing

5. Repeat **Insert Component** to place *pin.dwg* as the other component. Insert it next to the piston.



Pin added to the assembly drawing

6. With the two parts in the drawing, you can attach them to each other. This is done by employing 3D constraints. Working in 3D takes pre-planning, and so let's think through what is needed:
   - You want the pin to stay inside the piston head
   - The pin must be free to rotate, but it cannot slide in and out of the piston

   To accomplish this goal, you need to apply two 3D constraints:

   Concentric constraint keeps the pin centered inside the hole of the piston (but allows the pin to slide out of the piston)

   Tangent constraint keeps the pin from leaving the piston

   a. Open the 3D Constraints toolbar by right-clicking any toolbar and then choosing **BRICSCAD | 3D Constraints**.

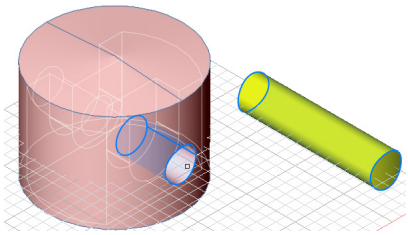
3D Constraints toolbar for attaching components

   b. Click **Concentric**:

```
: dmconcentric3d
```

   c. And then pick a curved face (a.k.a. subentities) from the piston and the pin:

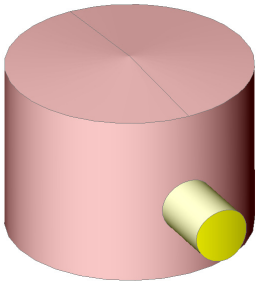
```
Select a pair of subentities: (Pick the curved face of the pin, highlighted in blue on the
yellow part shown in the figure below)
Entities/subentities in set: 1
Select a pair of subentities: (Pick the curved inside face of the piston, also highlighted
in but on the pink part)
```



Selecting curved surface to make components concentric

TIP Should you have difficulty picking the correct face with the cursor, press the **Tab** key to cycle through all possible surfaces under the cursor.

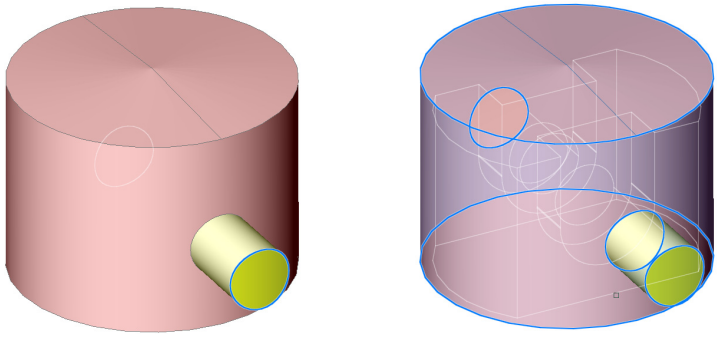
   The command ends automatically after you pick the second subentity. Notice that the pin jumps over to the opening of the piston. The pin is inside the piston; now you use the Tangent constraint to keep the pin from sliding out of the piston.



Concentric constraint lines pin up with piston's opening

d. To shove the pin inside the piston, making its ends flush with the piston walls, use the **Tangent** constraint and pick the two subentities described here:

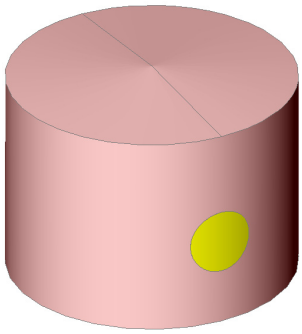
```
: dmTangent3d
Select a pair of subentities: (Pick one end of the pin; see blue outline in the figure below)
```



Left: *Selecting an end of the pin as the first tangent surface;* ***right:*** *Selecting the outside of the piston as the second tangent surface*

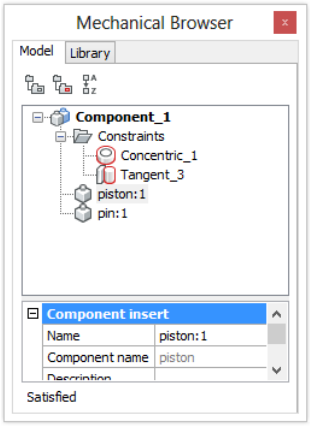
```
Entities/subentities in set: 1
Select a pair of subentities: (Pick the outside of the piston, shown outlined in blue in the figure above)
```

The constraint snaps the pin inside the piston.



Pin snug inside the piston

6. Look the content of the Mechanical Browser bar. It lists the two components (Piston:1 and Pin:1) and the two constraints used.



Mechanical Browser listing the components and constraints of this assembly

TIP To remove a constraint, right-click its name and choose **Delete**.

With the parts are attached to one another, they form an assembly. After this, simple kinematic analysis can be applied to the assembly, such as rotating and moving (sliding) parts. See section below. As well, the assembly drawing can be turned into 2D drawings and sections. Both of these tasks are described later in this chapter.

TIP Mechanical components are stored in *.dwg* files as custom objects. While they can be opened and viewed in AutoCAD, the constraints do not translate, because Bricsys and Autodesk use different code for constraints.

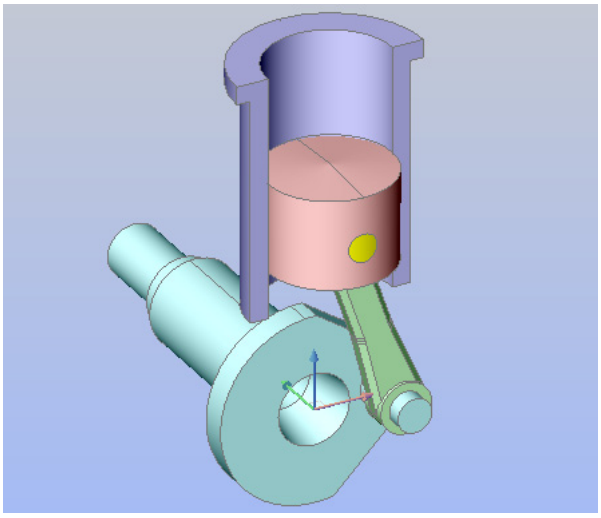
3D Kinematic Analysis

BricsCAD Platinum can perform two kinds of kinematic analyses, rotating or sliding parts held together in assemblies by 3D constraints. The analysis does not, however, perform collision detection. *Kinematic analyses* animates assemblies to show you how the parts move; *collision detection* determines if any of the moving parts would collide with one other. (AutoCAD has neither function.)

Kinematic Analysis Commands
dmRotate rotates entities and sub-entities
dmMove moves entities and sub-entities

DOING MOVEMENT ANALYSIS

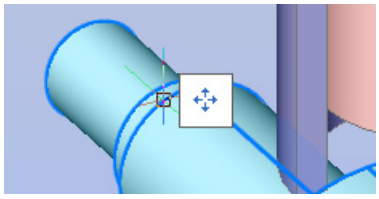
To see how kinematic analysis works in BricsCAD, open *Piston Engine.dwg*, a sample drawing provided with BricsCAD. (You'll find it in the *C:\Program Files\Bricsys\BricsCAD V17\en_US\Samples\Mechanical\Piston* folder.) This assembly drawing is complete, with all of the components held in place with 3D constraints. See figure below.



Sample drawing provided with BricsCAD

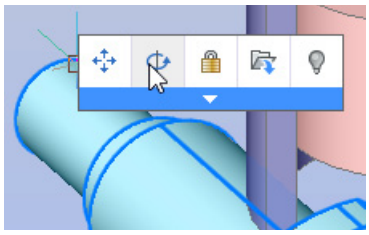
In this sample drawing, you rotate the parts of the mechanism with the **dmRotate** command. Start the command with the Quad cursor, like this:

1. Move the cursor over the crankshaft, and then wait a second for the Quad cursor to show up. Notice that the crankshaft is outlined in blue, which indicates the Quad cursor has selected it.



Quad cursor appears over selected entity

2. Move the arrow cursor over the single icon; notice that the Quad cursor expands to five icons.

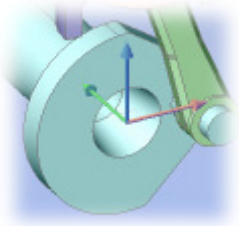


Selecting the Rotate command from the expanded Quad cursor

3. Choose the **Rotate** command.
4. Notice the prompt at the command line:

```
: dmRotate
Select axial entity or define axis by [2Points/Xaxis/Yaxis/Zaxis] <2Points>: y
```

Enter **y** for the y axis option. This is a clever shortcut, because the center of the crankshaft lies exactly along the y axis, as you can tell from the UCS icon in the figure below.

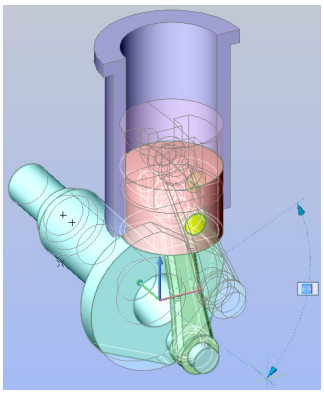


Crankshaft's centerline laying along the y axis

5. To start the rotation, pick a point anywhere in the drawing; the point you pick is not important, but further away from the y axis gives you finer control.

```
Pick start point in the rotation plane (Pick a point.)
```

6. Move the mouse to rotate the mechanism:



Crankshaft, link, and piston move together

Notice how the engine operates: as you move the mouse, you change the rotation angle of the crankshaft, causing all linked parts to rotate in tandem.

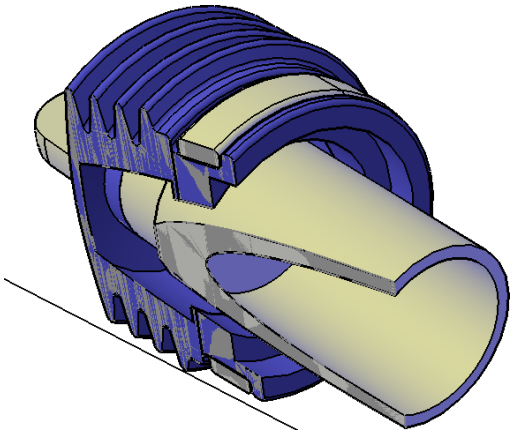
3D Sections

BricsCAD Pro and Platinum editions can make 2D and 3D sections of 3D models, and use the same commands as does AutoCAD:

Section Commands
Section creates section planes from 3D solids made of region entities
SectionPlane creates section entities from 3D solids, surfaces, and meshes
LiveSection toggles the Live Section property of a section plane
SectionPlaneSettings defines properties of section plane entity in the Drawing Explorer
SectionPlaneToBlock saves the selected section plane as a block

WORKING WITH SECTIONS

Sections in BricsCAD work exactly the same as sections in AutoCAD.

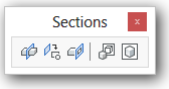


Live sectioning a 3D model in BricsCAD

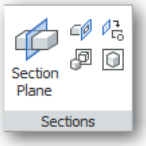
Accessing the Commands

To access the sections feature:

- Enter the commands listed in the table above
- Open the **Sections** toolbar



- In the ribbon's **Modeling** tab, look for the commands in the **Sections** tab



- From the **Model** menu, choose **Sections**

Drawing Views vs Model Documentation

BricsCAD Pro and Platinum editions generate 2D drawings and sections from 3D models. These are called "drawing views" (or "generative drawings" in earlier releases). Because the drawings are associative, they update automatically when you make changes to the 3D model. AutoCAD has the same function, but calls it "model documentation."

Generative Drawing Commands
ViewBase generates 2D views of 3D models in paper space
ViewDetail generates detail views from 2D views made by ViewBase
ViewDetailStyle specifies the style of detail views and detail symbols
ViewEdit changes the scale and the hidden line visibility of drawing views; can be used in paper space only
ViewExport exports generated drawings from paper space to model space; destroys 3D information
ViewProj generates additional projected views from existing drawing views
ViewSection generates sections from 2D views made by ViewBase
ViewSectionStyle specifies the style of section views

GENERATING DRAWINGS FROM MODELS

The method of placing 2D views of 3D models in BricsCAD is similar to that of AutoCAD:

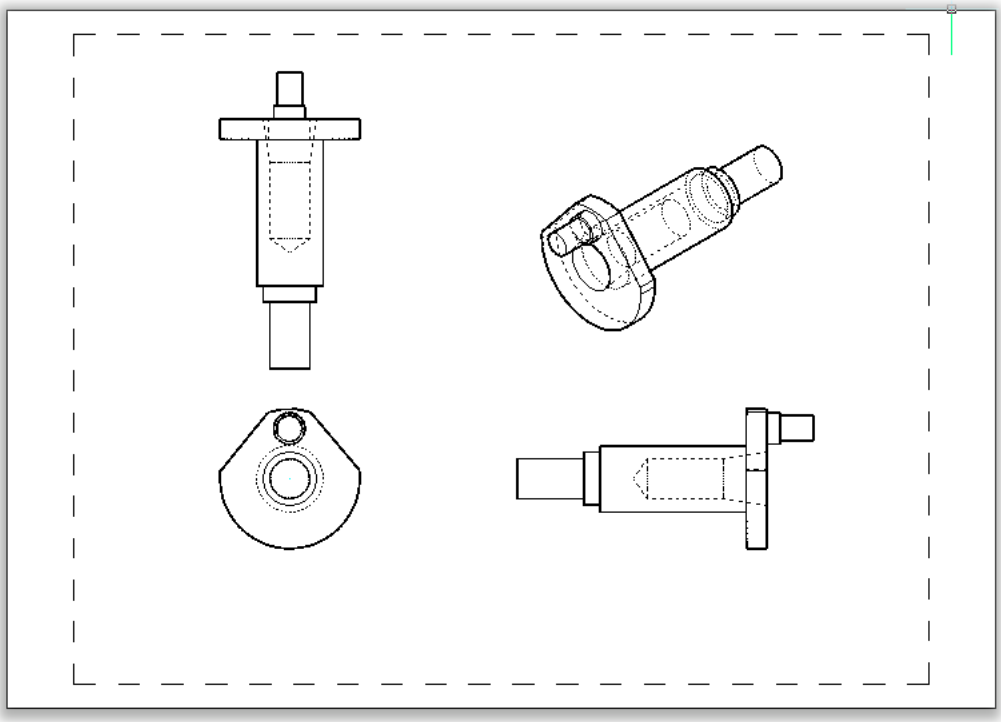
```
: ViewBase
Preset: "None",  View scale: "Adapt to paper size"
Select objects or [Entire model] <Entire model>: (Press Enter to select all)
Enter new or existing layout name to make current <Layout1>: (Press Enter to accept default)
```

BricsCAD switches to a layout automatically, and then you can start placing views.

The first view placed is the front view; other views are created automatically and depend on how you move the cursor.

```
Select position for base view [Scale/Tangent edges/Orientation/Projection type/Isometric
style/sElect] <Cancel>: (Pick a point to locate the first view, which is the front view)
Select position for projected view [Done] <Done>: (Keep picking locations for views...)
Select position for projected view [Done] <Done>: (...and then press Enter to exit the command.)
```

The result is a drawing that looks like this:



From top, clockwise: *the top, isometric, side, and front views*

Sections from Drawings

Creating sections happens like this: working in the same layout, you use the **ViewSection** command to place sections generated from the 2D drawings made earlier by the ViewBase command.

```
: ViewSection
Select drafting view: (Pick a view created earlier by the ViewBase command)
```

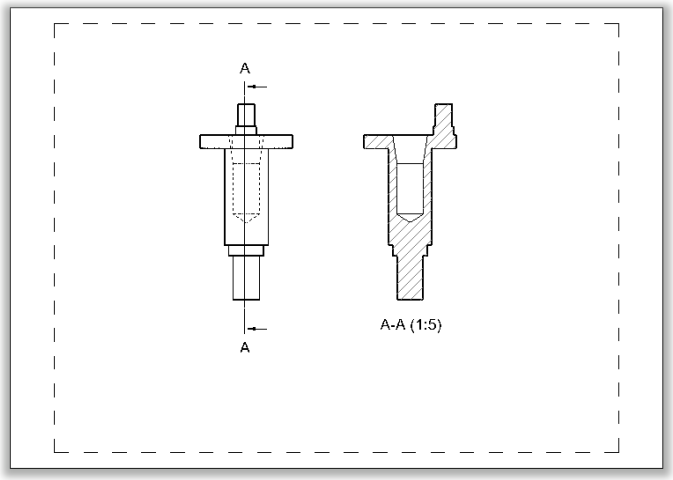
Now pick two points to become the start and end of the section line (A-A) that bisect the view:

```
Specify start point of section line or [Type] <Type>: (Pick a point at one end of the view)
Specify start point of section line or [Type] <Type>: (Pick the other point at the other end
of the view)
```

Finally, position the newly created section view:

```
Select position for section view: (Pick a point to the side of the view)
```

The result is a section view complete with cross hatching, section marker name, and scale factor. As of V17, you can make detail views from details.



Section view created by BricsCAD

Details from Drawings

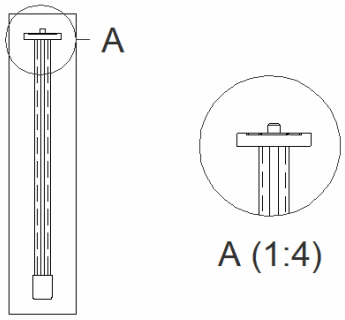
BricsCAD creates detail views through the **ViewDetail** command, as follows:

```
: viewdetail
Select drawing view: (Pick inside a drawing view; don't pick the viewport's border)
Specify detail center on source view: (Pick the point in the drawing view that you want to
be the center of the detail view.)
Select radius of detail view: (Indicate the extent of the drawing view.)
Select position for detail view [Scale] <Cancel>: (Pick a point to locate the detail.)
Select option [Scale/Hidden lines/Tangent lines/anChor/Annotation/Boundary/model Edge] <Can-
cel>: (Press Enter)
```

The default scale for the detail view is 1:4 (four times larger). Enter **S** to change the scale factor:

```
Select option [Scale/Hidden lines/Tangent lines/anChor/Annotation/Boundary/model Edge] <Cancel>: s
Adjust view scale [Standard scales/Custom scale/Relative custom scale] <Standard scales>:
Enter the scale number <1>: 2
```

Standard scales are provided by the ScaleListEdit command.



Detail view A

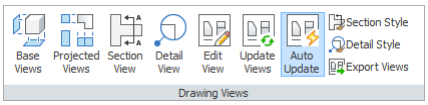
Section and Detail Styles

You can customize the way that sections and details appear with BricsCAD's **ViewDetailStyle** and **ViewSectionStyle** commands. These operate similarly to the way they do in AutoCAD.

Accessing Generative Drawing Commands

- Enter the one of the commands listed above
- Open the **Drawing Views** toolbar

- In the ribbon's Annotate tab, select commands from the **Drawing Views** panel



- From the **View** menu, open the **Drawing Views** submenu

Bills of Material vs Data Extraction

BricsCAD Platinum edition generates bills of materials from 3D models with its **bmBom** command. AutoCAD does the same through the DataExtraction command, which has the option to place the data as a table in the drawing. The difference is that the command in BricsCAD is easy to use (enter no options, if you wish), while the command in AutoCAD is very complex, and requires many steps.

HOW BMBOM WORKS

The BricsCAD bill of materials function works only with drawings created as assemblies and components. Open such a drawing, and then enter the **bmBom** command:

```
: bmbom
Insertion point [Name/Top level/Bottom level]: (Pick a point in the drawing, or enter an option)
```

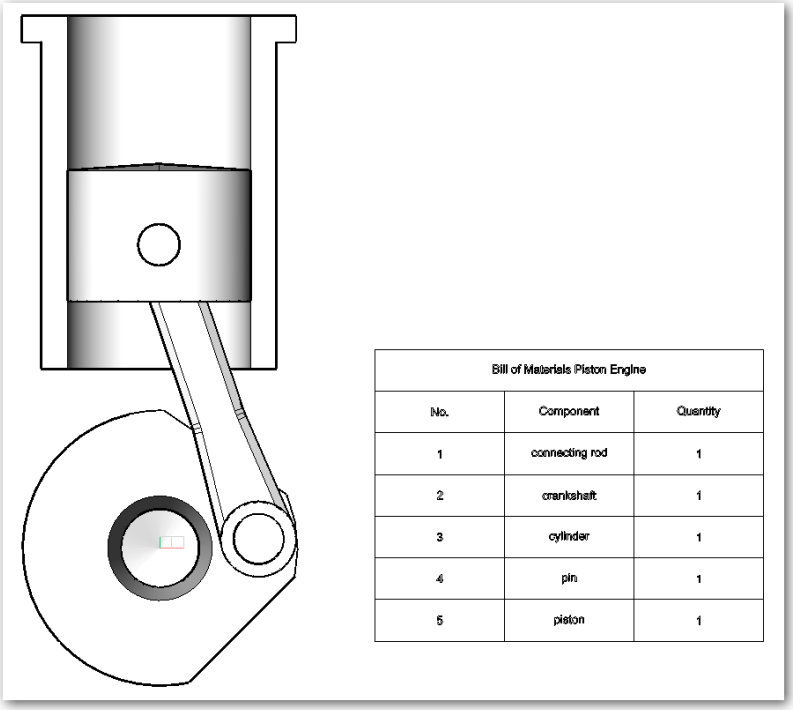
Bill of Materials Piston Engine		
No.	Component	Quantity
1	connecting rod	1
2	crankshaft	1
3	cylinder	1
4	pin	1
5	piston	1

Elements of a bill of materials

The bill of material table has a fixed format and lists mechanical components as follows:

No. is the components's serial number, and always begins with 1
Part identifies the name of the component, as extracted from the Mechanical Browser
Quantity reports the number of occurrences of each component

Among the command options, **Name** changes the title from the default, which is "Bill of Materials <drawing name>"; **Top level** or **Bottom level** determine which components are listed in the table.

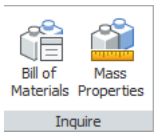


BOM table inserted in a drawing

TIP BOMs are normal table entities, and so their content and the tables' cells can be edited like a table. To export the data in the table to a data file, use the **TablExport** command.

Accessing the BOM Command

- Enter the **bmBOM** command
- Open the **Mechanical** toolbar, and then click the **Bill of Materials** button
- In the ribbon's **Assembly** tab, look in the **Inquire** panel

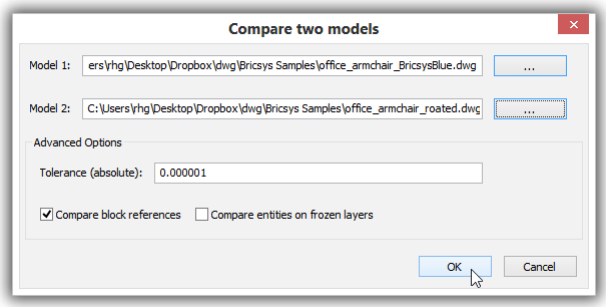


- From the **Mechanical** menu, choose **Bill of Materials**

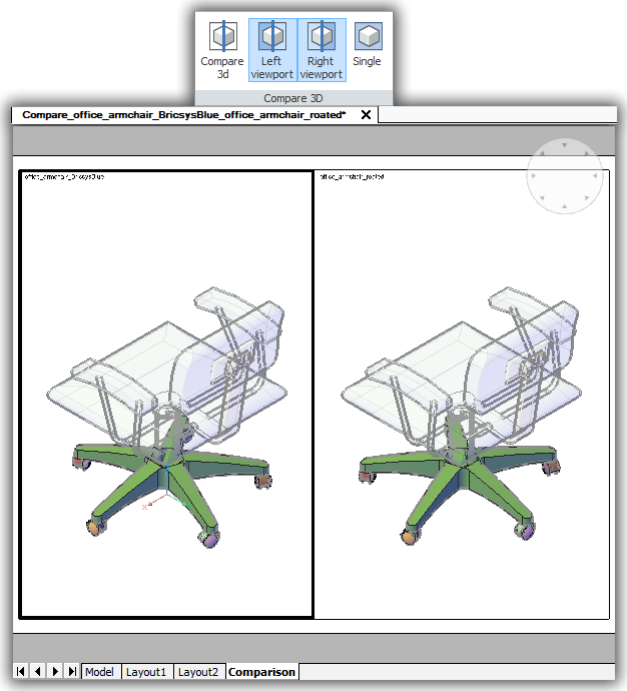
3D Compare

(new to V17) The **3dCompare** command loads two drawing files, and then finds differences among 3D solids and surfaces using color coding. (AutoCAD does not provide this capability.) In the ribbon, there are toggles that change what you see.

1. Enter the **3dCompare** command, and then choose two drawings files whose content you want to compare. Keep in mind that this commands compares differences only in 3D solids and 3D surfaces; it ignores all other entities, such as dimensions, text, and 2D entities.



2. Click the **Model 1** ... button to select the first drawing file.
3. Click the **Model 2** ... button to select the comparison drawing.
4. Click **OK**. Notice that BricsCAD opens both models in a new viewport named "Comparison." In the figure below, the base of the chair is colored, because it is different in the second drawing. (The base is rotated by 15 degrees from the original.)



Comparing two slightly different models

5. In the **Tools** tab, buttons in the **Compare 3D** panel let you toggle view settings.

CHAPTER SEVEN

BIM, Sheet Metal, & Communicator

BRICSYS OFFERS ADD-ON MODULES THAT PERFORM SPECIALIZED FUNCTIONS IN THE AREAS of architectural design, sheet metal fabrication, and translation to and from other 3D MCAD (mechanical CAD) systems. In this chapter, we look at the following add-ons available as a free 30-day trial from the associated Web page.

- BIM (building information modeling) for architects ($240) *
  https://www.bricsys.com/en_INTL/bim/
- Sheet metal design ($300) *
  https://www.bricsys.com/en_INTL/sheetmetal/
- Communicator export-import, including the import of assemblies ($610)*
  https://www.bricsys.com/en_INTL/communicator/

  * Functions missing from AutoCAD

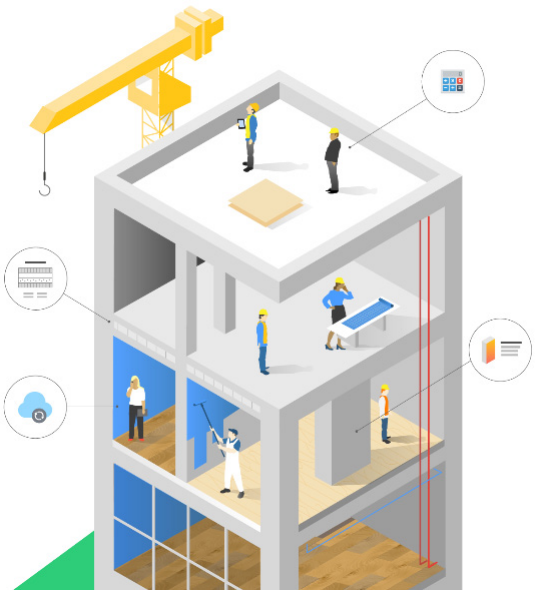
Each of these add-on modules is an extra cost, priced in the hundreds of dollars. Equivalent software from Autodesk is, however, in the thousands of dollars — except for the import-export module, which Autodesk provides its customers for free. Prices are shown in US$ and accurate at time of writing.

3D BIM Design

BricsCAD Platinum supports an optional add-on that models buildings in 3D using BIM (building information modeling). Any 3D solid can be used with the BIM model, whether created in BricsCAD or imported from other software. BricsCAD imports and edits BIM models from other CAD systems using the IFC format.

The BIM module provides commands specific to architectural design, and is available for purchase from . Commands shown in blue are new since the last edition of this book.

Building Information Modeling Commands
bimAttachComposition attaches BIM compositions (wall styles) to solids
bimCheck reports the number of BIM entities in drawings
bimClassify classifies entities as a wall, slab, column, beam, window, or door
bimConnect creates L-connections between two solids
bimDrag extends walls or slabs; modifies their thickness
bimExport exports the model to an .ifc file, which contains all 3D geometric and BIM-related data
bimFlip flips starting faces of compositions; mirrors inserts like windows and doors
bimGetStatisticalData reports statistics data of BIM objects in the current drawing
bimIfcImport imports IFC files
bimInsert inserts window and doors
bimList reports DXF-style data on BIM entities in drawings
bimReposition repositions inserts (doors, windows) in the faces of solids
bimRoom defines a room by clicking inside a room area or by selecting a 3D solid
bimSection creates sections from BIM models
bimSectionOpen opens drawing files related to BIM sections
bimSectionUpdate exports BIM sections; also updates BIM sections
bimSkpImport imports SKP SketchUp files with optional stitching
bimSpatialLocations opens the Buildings & Stories Manager for sites, buildings, and stories
bimSplit automatically separates segmented solids, or by selection of cutting faces
bimUpdateRoom executes room-finding algorithm to redefine rooms, such as if new walls are added
bimUpdateThickness reapplies overall thickness of compositions to solids
bimWindowArray places an array of inserts, such as windows and doors
bimWindowPrint imprints 2D window and door outlines into walls
bimWindowUpdate updates openings made by windows and doors when their definition changes



HOW BIM DESIGN WORKS

BIM designs commonly begin the terrain on which the building is to be situated, then the building is designed with one or more floors. BricsCAD can handle this, but for this tutorial, we'll do something a little simpler.

We begin with a 2D floor plan, and then extrude with the **PolySolid** command into walls and floors.

1. Start BricsCAD with the **BIM** workspace and **Imperial** units.

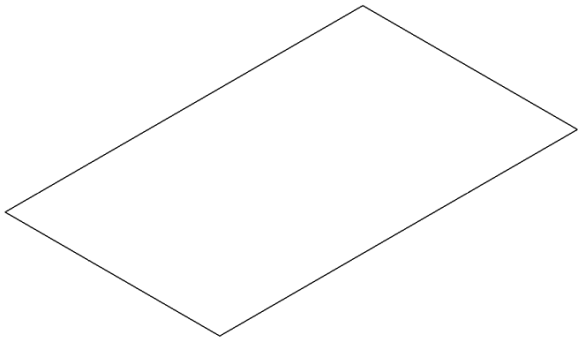
2. To make it easier to see your work, change the visual style **Wireframe**. You can do this in the Properties pane Explorer with the **View > VisualStyles** option, shown at right.

   ... or at the command prompt with the **-VisualStyles** command:

   ```
   : -visualstyles
   Visual styles: set_Current/Saveas/Rename/Delete/?: c
   Enter visual style [2dwireframe/Wireframe/Hidden/Realistic/Conceptual/Shaded/shaded with
   Edges/shades of Grey/SKetchy/X-ray/Other/cUrrent]: <Shades of Gray>: wireframe
   ```

3. Draw an outline of the floor plan. For this tutorial, draw a rectangle 50' by 25' with the **Rect** command's Distance option. This is the typical size of a house in North America.


```
: rect
Select first corner of rectangle or [Chamfer/Fillet/Rotated/Square/Elevation/Thickness/Width
of line/Area/Dimensions]: d
Length to use for rectangles <0">: 50'
Width to use for rectangles <0">: 25'
Select first corner of rectangle or [Chamfer/Fillet/Rotated/Square/Elevation/Thickness/Width
of line/Area/Dimensions]: 0,0
Other corner of rectangle: (Pick a point in the upper right corner of the drawing area)
```



Rectangle defining the floor area

4. With the **PolySolid** command, turn the floor plan into walls.


```
: polysolid
Current settings: Height = 80, Width = 5, Justification = Center, Separate solids = On, Dy-
namic = On
```

a. To make it quicker to use, preset the values:

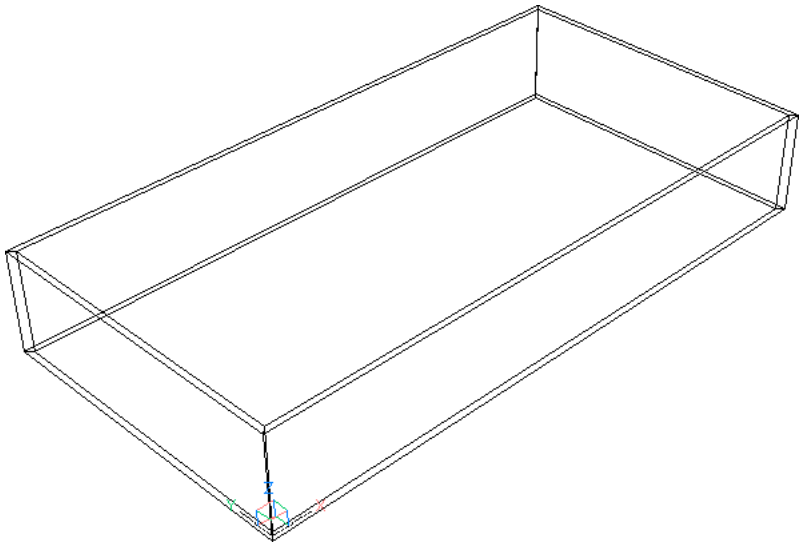
PolySolid Option	Value	Notes
Dynamic	Off	Prevent command from prompting for heights and widths
Height	8'	Typical floor to ceiling height
Width	6"	Typical width of exterior walls; use 4" for exterior walls

```
Start point or [Height/Width/Justification/Entity/Separate solids/Dynamic] <Entity>: d
Dynamic height On/OFF <On>: off
Start point or [Height/Width/Justification/Entity/Separate solids/Dynamic] <Entity>: w
Width of polysolid <80>: 6”
Start point or [Height/Width/Justification/Entity/Separate solids/Dynamic] <Entity>: h
Height of polysolid <5>: 8’
```

b. Now you’re ready to apply the command to the rectangle. Enter the **Entity** option, and then pick the rectangle:

```
Start point or [Height/Width/Justification/Entity/Separate solids/Dynamic] <Entity>: e
Select polysolid base: (Pick the rectangle)
```

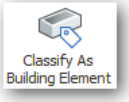
Notice that the walls appear instantly.



PolySolid command raising the walls

c. If you don’t see all of the walls, use **Zoom E** to zoom the drawing to the extents.

5. The next step is to tell BricsCAD that these are walls. You do this with the **bimClassify** command.




```
: bimclassify
Classify entities as [Wall/Column/Slab/Beam/wIndow/Door/building Element/Other/Auto/Unclas-
sify]: w
Select entities to classify: all
Entities in set: 4
Select entities to classify: (Press Enter to finish)
BIM data assigned to 4 object(s)
```

6. With the walls in place, the next step is to define their *composition* — what are the walls made of? Here is the composition of typical walls in homes of North America:
   - **Exteriors** of walls consist of exterior and interior *cladding* that give walls their look. Cladding is made from bricks, wood, gyproc (drywall), and so on.

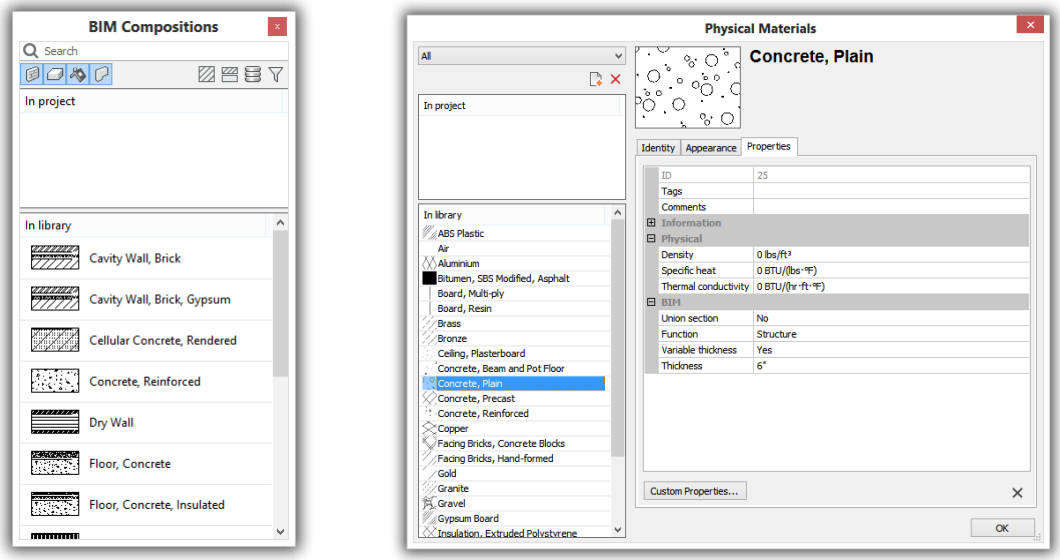
Tyvek in white and brick cladding in brown

   - **Interior** of walls provides strength through 2"x4" (interior walls) or 2"x6" (exterior) *studs* made of wood or metal. The strength of walls is needed to hold up walls, roofs, and so on. Extra pairs of 2"x6"- or 2"x10"-sized beams, called headers, are needed over window and door openings to distribute weight.
   - **Between** the studs is insulation that retains the building's heat in winter and keeps out heat in summer. Depending on local construction bylaws, Tyvek-style wrap may be needed to keep out moisture and wind. The photo shows the white Tyvek wrap, along with some brick exterior cladding.
   - Also between the walls are utilities, such as electrical wiring and plumbing, but these are not defined by compositions.

The *composition* of walls in BricsCAD is defined through the **bimAttachComposition** command. You attach "compositions" to walls, floors, and roofs to define what they are made of:

- To use 40 or so compositions provided by BricsCAD, use the BIM Compositions panel; access it by right-clicking any toolbar or the ribbon, and then choosing **BIM Compositions** from the shortcut menu. See figure at left, below.
- To define your own materials and edit existing ones, use the Physical Materials dialog box (formerly named the Building Materials dialog box. Access it by clicking the **Materials** button in the BIM Compositions panel. See figure at right, below.

- To combine materials into compositions, use the Compositions dialog box. Here you take one or materials and then layer them into a composition, such as brick-tyvek-plywood. Access it by clicking the **Compositions** button in the BIM Compositions panel.



***Left:** Pre-defined materials available in BricsCAD; **right:** dialog box for customizing material*

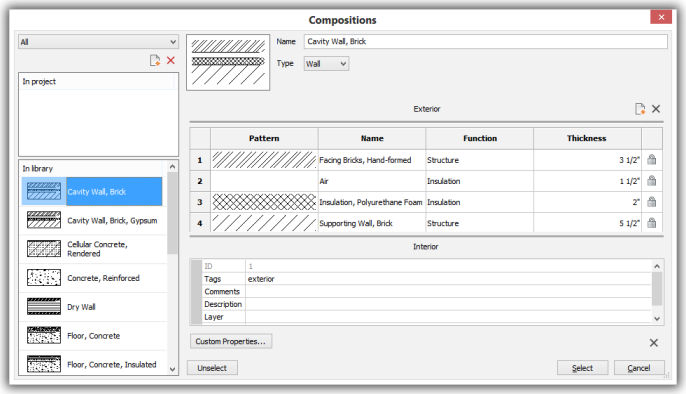
TIP The easy way to get to the dialog boxes that define materials and compositions is by clicking their buttons in the BIM Compositions panel:

***Buttons, left to right:** Materials, Compositions, Project and Library, Filter*

For this project, apply the "Cavity Wall, Brick, Gypsum" composition to all walls at once, as follows:


```
: bimattachcomposition
Select entities to attach composition: all
Entities in set: 4
Select entities to attach composition: (Press Enter to continue)
Enter composition name or [Dialog] <Dialog>: d
```

Notice the Composition dialog box. Choose "Cavity Wall, Brick, Gypsum" and then click **Select**.



Selecting a composition for the walls

```
The composition has been assigned to 4 element(s).
```

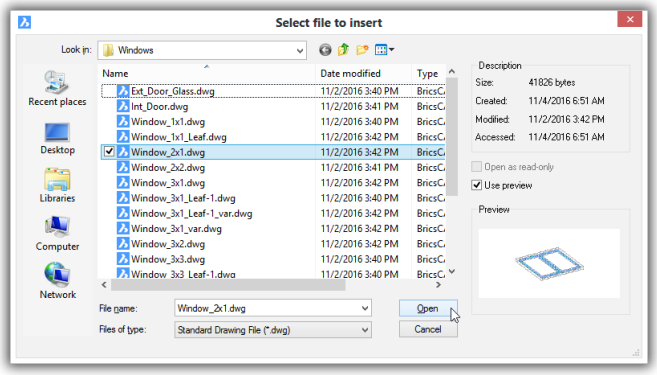
The walls look no different, and changing the visual style doesn't show the bricks either, because this is data that has been applied. The BIM Compositions panel, however, lists the composition you applied.

7. With the walls set up, add a window with the bimInsert command. You can use any block for this, although BricsCAD includes with a selection of them. Enter the **bimInsert** command:

: `biminsert`

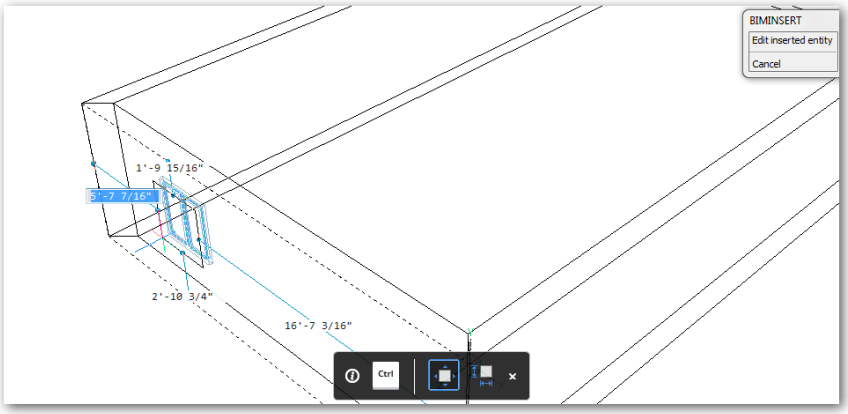
a. Notice the Select File to Insert dialog box. Choose a window block, such as "Window_2x1.dwg".

b. Click **Open**.



Choosing a window type

c. Position the window block over one of the walls. Notice that dynamic UCS kicks in to force the block to be coplanar with the wall you select.



Dynamic dimensions positioning the window, with Tips bar in black

Also kicking in are dynamic input (the dimensions that appear in the drawing area) and the Tips bar. When you press **Ctrl**, the Tips bar changes the command between **Insert** and **Edit** modes:

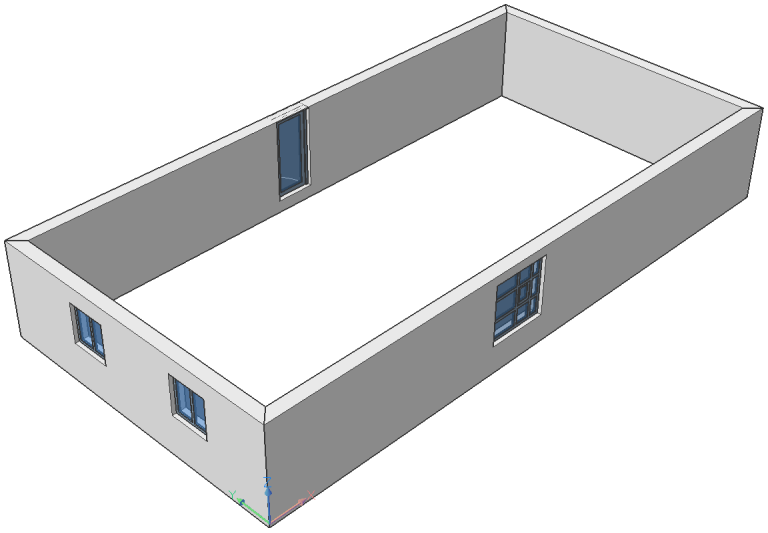
Icon	Meaning
	**Insert** dynamically dimensions the location of the window in the wall; prompts: `Select insertion point or [Edit inserted entity]:`
	**Edit** — allows you to change the size of the window; prompts: `Edit Height [Width/Done]:`

Press **Tab** to move between the dimension fields.

d. For this tutorial, just insert the window anywhere in the wall:

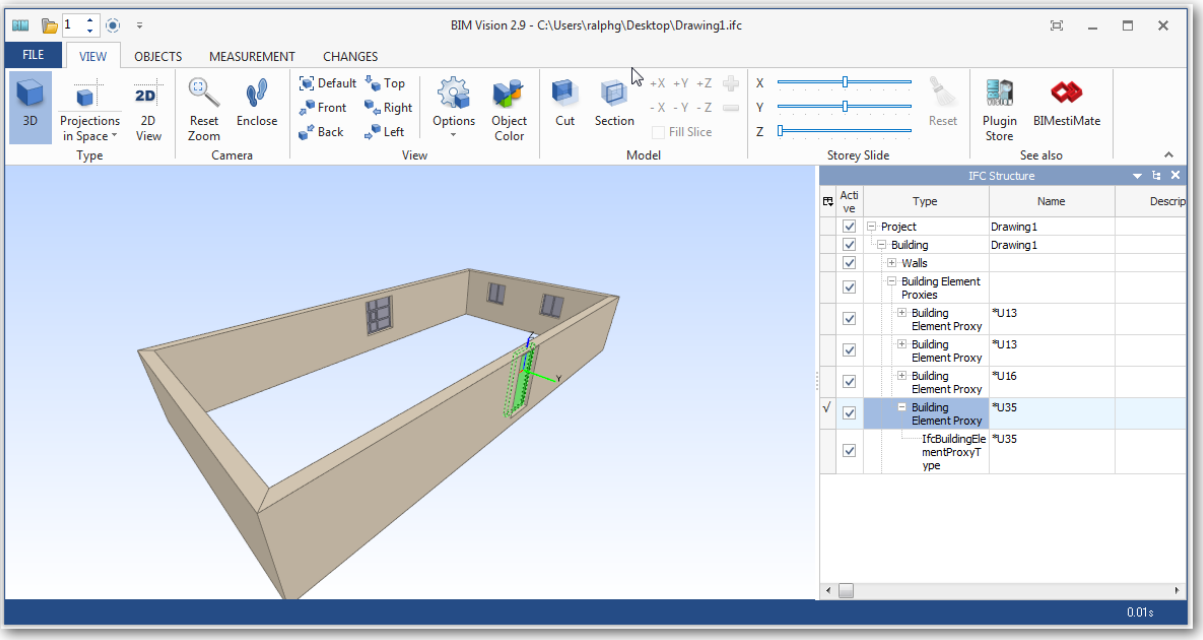
```
Select insertion point or [Edit inserted entity]: (Click to place the window)
```

e. Repeat the **bimInsert** command to place more windows and even a door. To see a nicer rendering of the building, change the visual style to "BIM."



Placing windows and a door

8. To export the model in IFC format, use the **bimExport** command.
9. To view the IFC file, use an IFC file viewer, such as the free one from http://bimvision.eu/en/download.

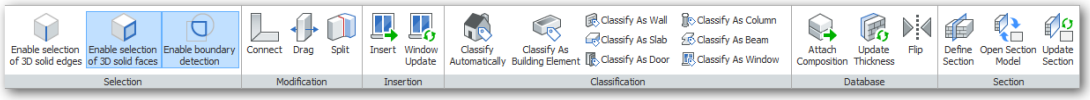


Viewing IFC data with a viewing program

Accessing BIM Commands

- Enter one the commands listed above
- Open the **BIM** toolbar

- In the ribbon's BIM tab, choose a command



- From the **BIM** menu, choose a command

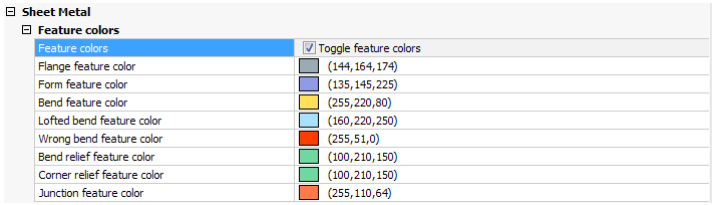
3D Sheet Metal Design

BricsCAD Platinum creates, bends, and unbends sheet metal designs with the Sheet Metal add-on.

Sheet Metal Commands
smBendCreate converts sharp edges between flange faces to bends
SmBendSwitch converts bends to lofted bends
smConvert recognizes flanges and bends in a 3D solids automatically
smDelete removes junctions by restoring sharp edge between two flanges
smDissolve dissolves sheet metal features
smExport2D exports sheet metal as unfolded representation of 2D profiles in .dxf or .dwg format
smExportOsm export a sheet metal designs in Open Sheet Metal .osm format
smFlangeBase creates sheet metal models from closed 2D polylines or regions
smFlangeBend command bends existing flanges along lines, obeying the k-factor for given bend radius
smFlangeConnect closes gaps between two flanges; their orientation does not matter
smFlangeEdge bends the sheet metal to make flanges; generates corner and bend reliefs automatically
smFlangeRotate changes the bend angle of flanges
smFlangeSplit splits flanges along a line drawn on their faces
smForm command converts a selected set of faces to form features
smJunctionCreate converts hard edges into junctions
smJunctionSwitch changing symmetrical junctions to ones with overlapping faces
smLoft constructs sheet metal bodies with lofted bends and flanges
smReliefCreate creates proper corner and bend reliefs
smReplace command replaces form features with ones from built-in or user libraries
smRepair joins connected lofted bends surrounded by flanges, rebuilds them tangent to adjacent flanges
smSelect command selects hard edges, same and similar form features
smSelectHardEdges selects all hard edges, and then reports about them in the report panel
smUnfold unfolds sheet metal bends

THE COLOR OF SHEET METAL

BricsCAD uses a color coding system to identify features in sheet metal parts.The colors listed below are found in the Settings dialog box. Bends are shown in yellow, for example, while corner reliefs (openings) are bright green.



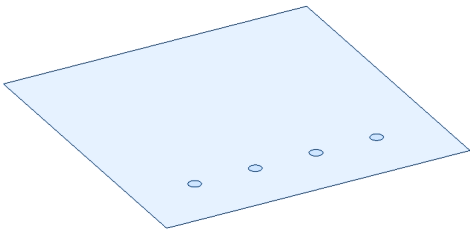
To turn off the coloring system, change the value of the **FeatureColors** variable to Off.

If you want to change the colors, go into the Settings dialog box and then use the Search field to look for "feature colors."

TUTORIAL I: HOW SHEET METAL DESIGN WORKS

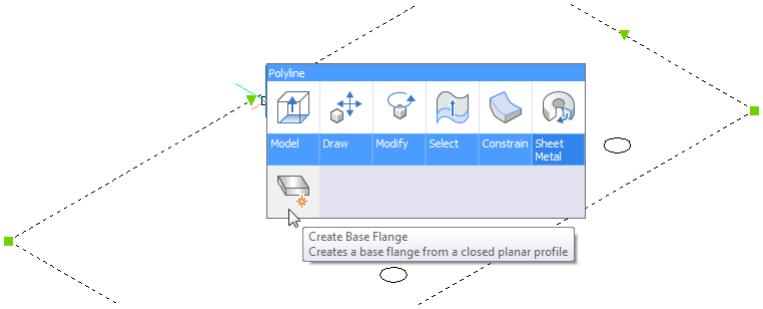
Sheet metal design begins with 2D profiles or 3D models, includes those imported into BricsCAD from other MCAD systems. This tutorial takes you through the fundamental steps using a 2D profile:

1. Start BricsCAD.
2. Draw a shape with a closed polyline or region:
   a. Draw a rectangle with the **PLine** command
   b. Add four openings with the **Circle** command
   c. Convert all five entities into a single region entity with the **Region** command



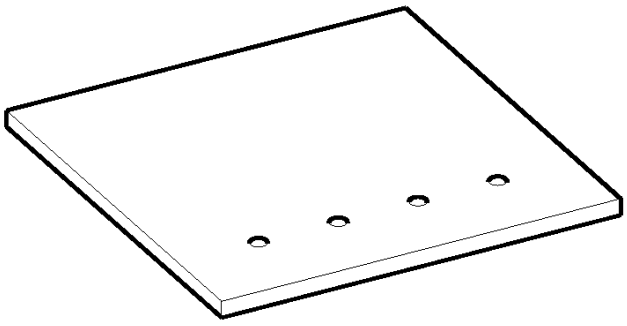
Rectangle and four circles converted to a region entity

3. Use the Quad cursor to start the **smFlangeBase** command by pausing the cursor over the region entity:



Using the Quad cursor to start the smFlangeBase command

When you click the **smFlangeBase** button, BricsCAD instantly turns the region into a sheet metal object. Notice that the region thickens. The object is now a 3D solid that BricsCAD recognizes as a sheet metal object.

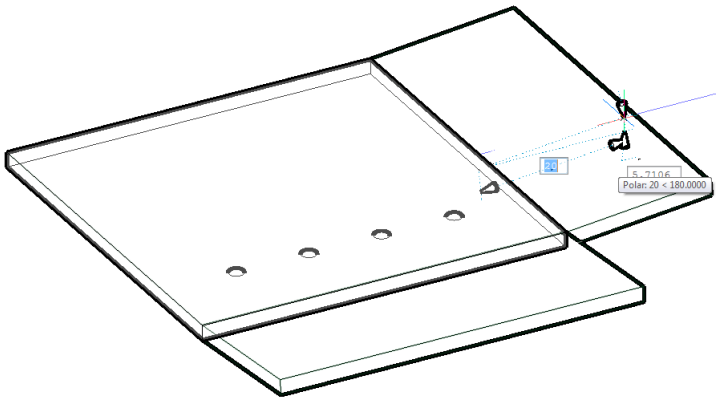


The smFlangeBase command thickens the region

4. To create sides (*flanges* that are pulled from the base), apply the **smFlangeEdge** command:

```
: smFlangeEdge
Select edges on flanges: (Pick an edge)
Entities: 1
Select edges on flanges: (Pick an adjacent edge)
Entities/subentities in set: 2
Select edges on flanges: (Press Enter to end edge selection)
```

Notice that BricsCAD *adds* sides (flanges) to the existing base; it does not subtract them. You specify the height of the sides in the next step.

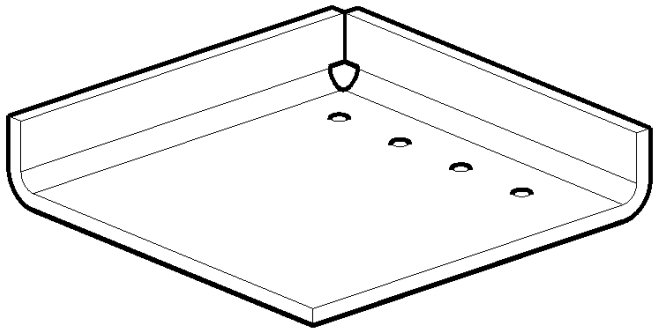


Two edges selected to bend

5. Move the mouse to indicate the angle of the bend, or else enter values at the keyboard for angle or length.

```
Position the end of the flange [Angle/Length/Taper angle/Width]: (Move the mouse to indicate
the angle, or enter values)
Position the end of the wall [Angle/Length]: a
Enter bend angle <Back>: 90
Position the end of the wall [Angle/Length]: l
Enter length of wall <Back>: 10
Position the end of the wall [Angle/Length]: (Press Enter to end the command)
```

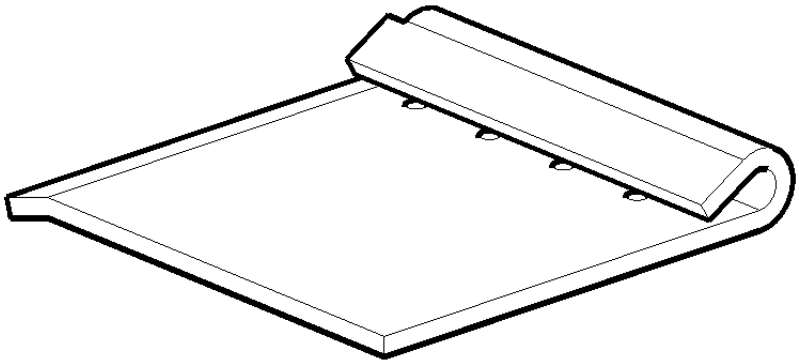
Notice that this command adds bends, bend reliefs, and corner reliefs automatically.



Sides bent into place

6. Should you wish to change the angle of a flange, use the **smFlangeRotate** command. Pick a face on the flange to be re-bent, as follows:

```
: smFlangeRotate
Select a flange face to rotate: (Pick a face -- not an edge! -- and then move the mouse to
show the new angle)
```

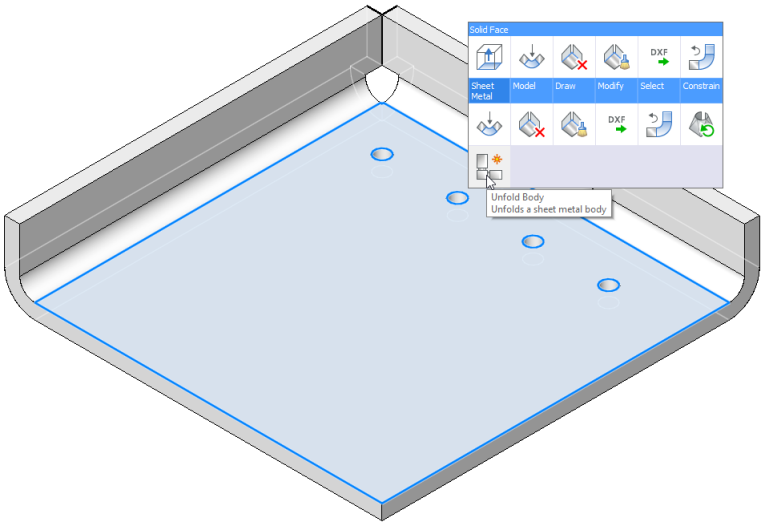


Changing the angle of flanges

TIP You can use any of BricsCAD's direct modeling and 3D constraints commands to edit sheet metal parts. In addition, you can control parts with user-defined parameters, such as material thickness and bend radius.

7. Designs are unfolded with the **smUnfold** command. The command is like the flatten command of other sheet metal programs. This command performs two jobs: it generates a 2D drawing of the sheet metal part, and then optionally exports the drawing in DXF format for use with CAM (computer-aided manufacturing) systems of sheet metal parts.

   Start the command from the Quad cursor:



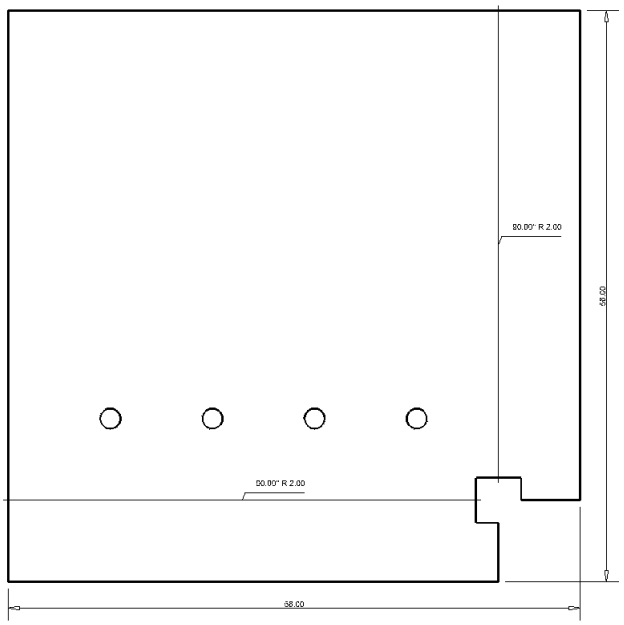
Accessing the smUnfold command

: **smUnfold**

Select a flange or lofted bend face to start unfolding [Settings]: *(Pick a point to place the 2D drawing)*

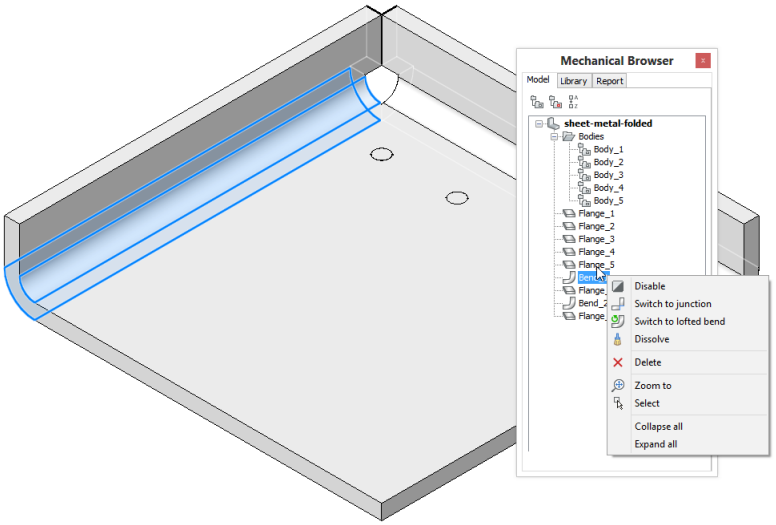
Select position of the unfolded body: *(Pick a point in the drawing away from the 3D model)*

Validate the unfolded body and select an option [save 2D geometry/save 3D geometry/Optimize bend annotations/Keep] <Keep>: *(Enter an option; see table below)*



Annotated 2D drawing of the sheet metal part

Use the Mechanical Browser to access the parts of the sheet metal part:



Clicking a node in the browser highlights the related part in the model

TIP To make the holes, use the **dmExtrude** command on the four circles.

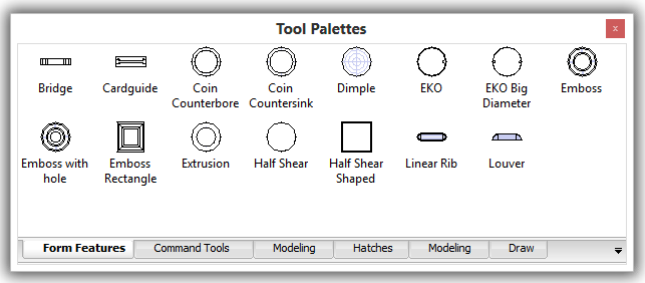
Because BricsCAD Platinum features design intent, you need to only extrude the one hole; BricsCAD recognizes the other three as having the same diameter, and so turns them into holes automatically!

Adding Form Features

(NEW TO V17) Forms are parts commonly added to sheet metal designs, such as louvers and embossed holes. BricsCAD provides them in a library so that like other blocks you don't need to draw them repeatedly. Forms are provided as 3D parametric parts, and are found in the Tool Palettes panel's Form Features tab.

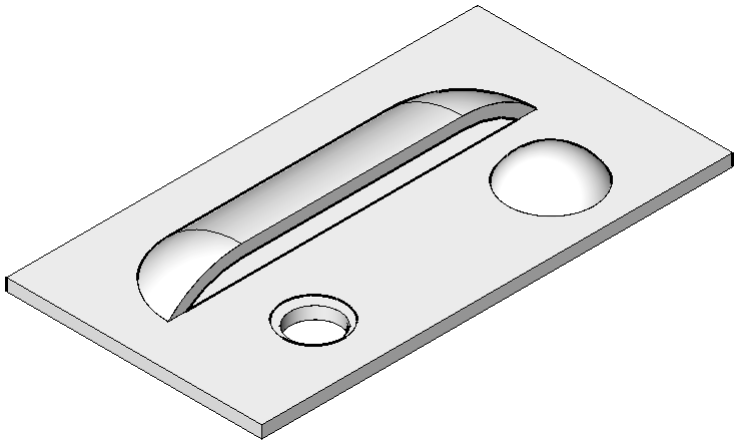
You use them like this:

1. Open the Tool Palettes with the **ToolPalettes** command, and then click the Form Features tab.



Form features found in the Tools Palette panel

2. Drag a feature onto the sheet metal piece. After it is placed, you can still move it.



Louver, countersink, and dimple placed on sheet metal

To control the appearance of form features in 2D and 3D unfolded model representations, change the value of the smFormFeatureUnfoldMode variable; it must be modified through the Settings dialog box at time of writing.

smFormFeatureUnfoldMode	Meaning
0	Keep
1	Remove
2	Project
3	Contour
4 (default)	Symbol

BricsCAD recognizes form features in geometry imported from other CAD systems. BricsCAD stores the features as individual .dwg files in the following folder: *C:\Users\userid\AppData\Roaming\Bricsys\BricsCAD\V17x64\en_US\Support\DesignLibrary\SheetMetal\FormFeatures*.

Exporting Sheet Metal Parts

CNC machines typically read DXF files to produce parts. Use the **smExport2D** command to export sheet metal designs as 2D profiles in *.dxf* format to as far back as Release 9. The **smTargetCAM** system variable specifies the CAM system to which to export.

TUTORIAL II: FROM 3D SOLID TO SHEET METAL

The above tutorial showed you how to create a sheet metal part from scratch. This approach is best for simple parts. BricsCAD, however, has a second approach: it can also create sheet metal models from 3D solids, which is a better approach for complex parts. MCAD programs like Solid Edge and Solidworks also have the ability to convert 3D solids into sheet metal parts. In this area, BricsCAD has a distinguishing feature, because the other two MCAD programs make the same mistake: the basic feature is an inseparable flange+bend, whereas in BricsCAD flanges and bends are independent.

This means that for most changes, users of those other two MCAD programs must restart from scratch; furthermore, they cannot split the model in several bodies, something that can be required when working with sheet metal designs.

TIPS Note that this tutorial works in BricsCAD only when it is the Platinum edition and when you have purchased the Sheet Metal add-on module from https://www.bricsys.com/en_INTL/sheetmetal/.

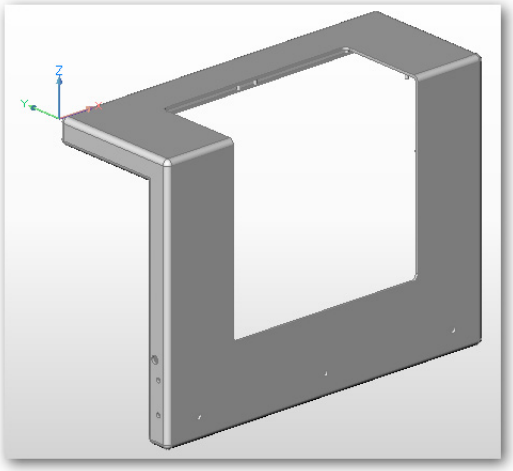
To import 3D models from other CAD packages, BricsCAD Platinum must be running Communicator, an optional, extra-cost file translator available from https://www.bricsys.com/en_INTL/communicator/. In BricsCAD, start a new drawing, and then enter the **Import** command to select the file to import.

In this tutorial, you defeature an solid model, and then convert it to a sheet metal part. *Defeaturing* means removing parts that can't be used in sheet metal stamping such as pins, or that need to be replaced, like fillets with bends.

Defeaturing is done with the assistance of two functions, *smart selection* and *subtraction extrusion*.

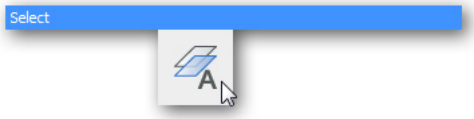
- "Smart selection" is useful by selecting all similar parts through the **dmSelect** command: you choose one feature, such as the face of a peg, and it selects all other identical faces in the mode. T
- "Subtraction extrusion" is when you remove the pegs by subtracting them with the direct modeling version of the Extrude command, **dmExtrude**.

1. Start BricsCAD in the **Sheet Metal** workspace, and then open the sample file *startfromsolid.dwg*.



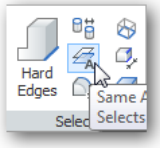
Solid model with pins and filleted corners

2. Here is the first step of defeaturing, smart selection. While you could perform smart selection at the command prompt, it is much easier using one of these icons:
   - From the Quad cursor, choose **Select** > **Same Area Faces**



Choosing the Same Area Faces command from the Quad

   - Or, in the Sheet Metal ribbon's **Select** panel, click the **Same Area Faces** button

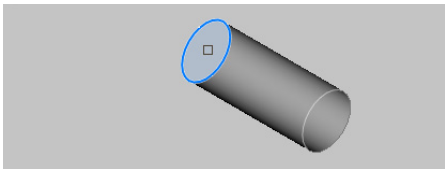


Finding the Same Area Faces button on the ribbon

3. Ignore the plural nature of the prompt by selecting the face of just one pin:

```
Select [sUbset/Sample/sEed] <Sample>: (Press Enter to accept the default, Sample)
Select several entities/subentities: (Pick the face of a pin)
```

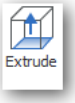
   Make sure that you select the *face*, and not the *edge*. (If you select the edge of the pin, then BricsCAD selects all other edges in the model, which you don't want.)



Selecting the face of one pin...

   Notice that BricsCAD selects all other faces that are the same.

4. With the pin faces selected, use the dmExtrude command to remove the pins, as follows:



a. Again, I recommend using the Quad or ribbon, as they automate some of the options you would have to otherwise specify at the command prompt. From the Sheet Metal ribbon's **Edit** panel, choose **Extrude**. Notice that BricsCAD fills in the first two prompts for you:

```
: dmExtrude
        Select entities/subentities to extrude or set [MOde]: _MO
        Choose type of created entity [SOlid/SUrface] <Solid>: _SO
```

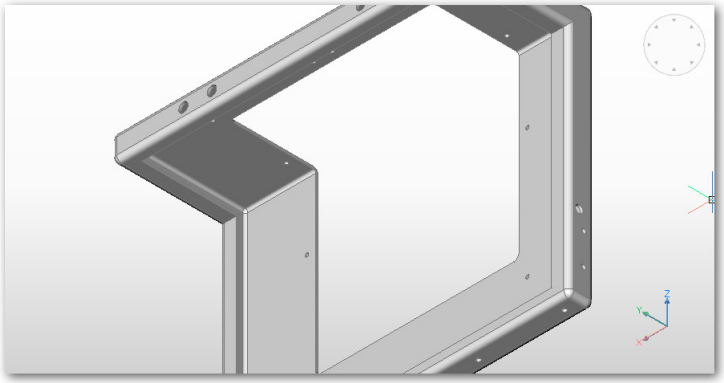
b. Specify 's' for the Subtract option:

```
        Specify height of extrusion or set [Auto/Create/Subtract/Unite/Taper angle/Limit]
        <Create>: s
```

c. Press **Enter** to end the command:

```
        Specify height of extrusion or set [Auto/Create/Subtract/Unite/Taper angle/Limit]
        <Subtract>: (Press Enter to end the command.)
```

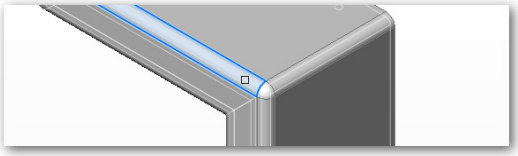
Notice that all of the pins in the model disappear instantly. They are replaced by holes, which will be stamped during the sheet metal manufacturing operation, after which pins are added separately.



Pins removed from solid model

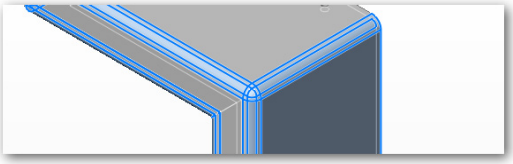
5. The other preparatory step is to remove the fillets so that the edges can later be turned into bends. Again, it is a two-step process: first select all fillets with dmSelect, and then erase them with the dmDelete command.

a. From the Sheet Metal ribbon's **Select** panel, choose the **Same or Less Radius Fillets** icon.

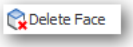


Selecting a fillet

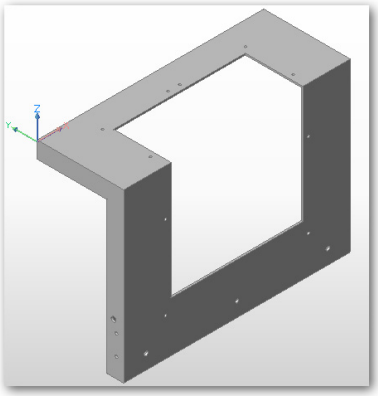
b. Choose a fillet. Notice that BricsCAD selects all the other fillets on the model, as shown in blue in the figure below.



All fillets selected in the model

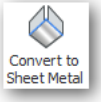


c. From the Sheet Metal ribbon's **Edit** panel, click **Delete Face**. Notice that all corners become sharp.



Fillets removed from the solid model

6. With the solid model defeatured, you now convert it to a sheet metal part with the smConvert command.



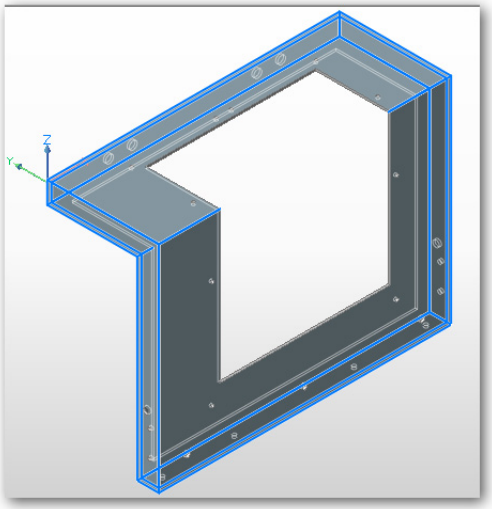
From the Sheet Metal ribbon's **Create** panel, choose **Convert to Sheet Metal**.

```
: smConvert
Select 3D solids/<Entire model>: (Press Enter to select the entire model)
```

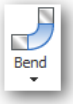
At the prompt, pressing **Enter** selects the entire model. The model looks no different, except that it takes on a gray color. From now on you edit it with commands that start with 'sm', short for sheet metal.

7. Convert all hard edges to bends. *Hard* edges are the ones with sharp edges. This process takes two steps: firstly, select all hard edges with the smSelectHardEdges command, and then turn them into bends with the smBend command. Here are the steps:

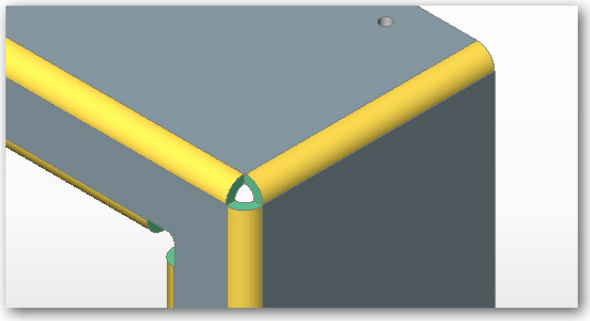
a. From the Sheet Metal ribbon's **Select** panel, click on **Hard Edges**. Notice that all hard edges are selected by BricsCAD, because they turn blue.



All hard edges selected by BricsCAD



b. Change the hard edges to bends. From the Sheet Metal ribbon's **Modify** panel, click **Bend**. Notice that the hard edges are replaced by bends, complete with cutouts at intersections. The bends are colored so that you can distinguish them visually from other sheet metal features.

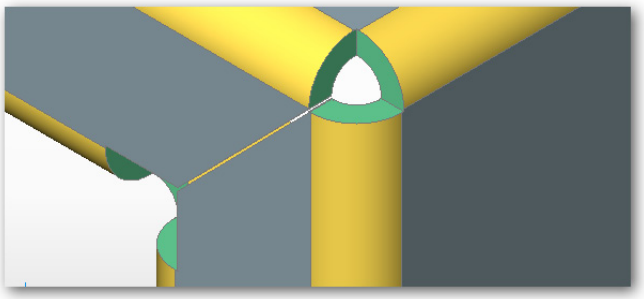


Bends (in yellow) complete with cutouts (in green) at intersections

8. The ultimate aim of sheet metal design is to produce a part that can be fully flattened, and so you need to fix up some corners manually by splitting flanges with the smFlangesplit command. Here's how:
   a. Zoom into a corner for a closer look with the **Zoom Window** command.
   b. Make sure that esnaps (entity snapping) are turned on. If necessary, click the **ESNAP** button on the status bar.

   c. From Sheet Metal ribbon's **Modify** panel, click the **Split** button. Follow its prompts on the command line:

```
: smFlangesplit
Select a flange face: (Pick a face)
Select lines, edges to split the flange or draw a <New line>: n
Start point of the line: (Use ensap to pick one corner; see figure below)
End point of the line: (Use ensnap to pick the other corner)
Make split Center/Left/Right/<Accept model>: (Press Enter to end the command)
```



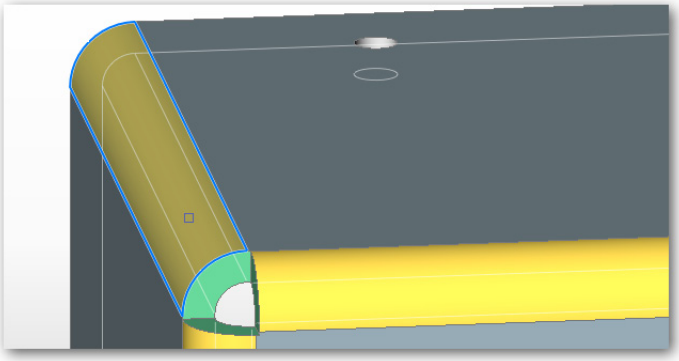
Splitting a flange

   d. Repeat for the other faces that need splitting.

9. A few other corners need to be turned into junctions. This is done with the smJunctionCreate command, as follows:

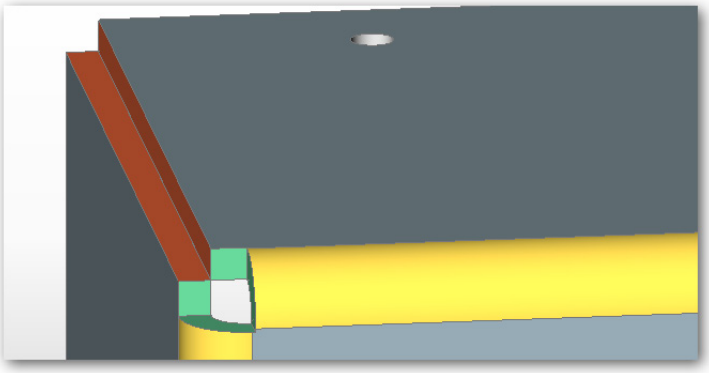
a. From the Sheet Metal ribbon's **Modify** panel, click **Junction**.

b. Pick a yellow-colored bend, such as the one outlined in blue, below.



Selecting a bend (outlined in blue)...

c. Notice that the bend immediately turns into a junction colored red. The command repeats automatically so that you can turn other bends into junctions. Continue making the change as required.



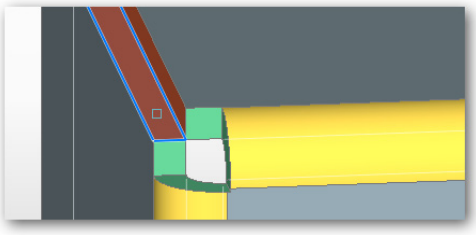
...and turning it into a junction (shown in red)

10. The junction needs to be edited so that one edge cleanly meets the other. You do this with the smJunction-Switch command, as follows:

a. From the Sheet Metal ribbon's **Modify** panel, choose the **Junction Switch** button.

b. Select one of the red faces, and then press **Enter** to end the command:

```
: smJunctionSwitch
        Select junction(s) face(s): (Pick one red face, as shown below)
```

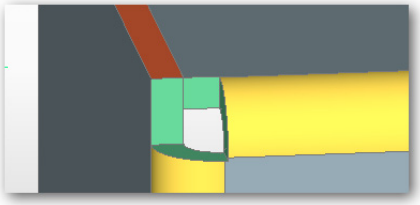


Selecting a face (in red)...

c. Press **Enter** to end the command.

```
Entities in set: 1
Select junction(s) face(s): (Press Enter to end the command)
```

Notice that BricsCAD extends one face to meet the other one automatically, as shown below:



..to make the edges match perfectly

d. Repeat for other junctions that need to be switched.

11. With the solid model properly prepared as a sheet metal part, it can be unfolded — the last step necessary before it is exported as a DXF or other file for stamping by CNC machinery. Unfolding is done with the smUnfold command.

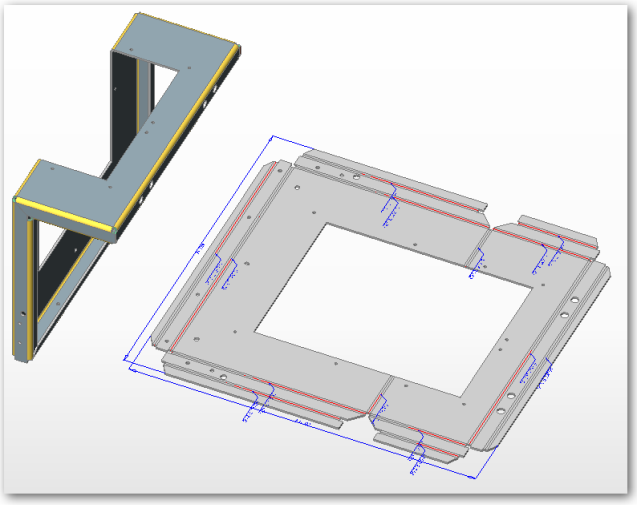
a. From the Sheet Metal ribbon's **Flatten** panel, choose the **Unfold Body** button. At the prompt, just pick any point on the sheet metal body:

```
: smUnfold
Select a flange or lofted bend face to start unfolding [lofted bend
        Tolerance]: (Pick a point on the body)
```

b. Pick a point in the drawing to place the unfolded sheet metal, and then press **Enter** to end the command:

```
Select position of the unfolded body: (Pick a point in the drawing)
Validate the unfolded body and select an option [save 2D geometry/save 3D
geometry/Keep] <Keep>: (Press Enter to end the command.)
```

Notice that BricsCAD automatically dimensions the flat part.

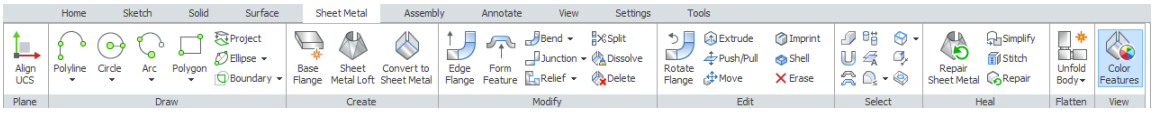


3D model flattened, ready for export to CNC machinery

Accessing Sheet Metal Commands

- Enter one the commands listed above
- Open the **Sheet Metal** toolbar

- In the ribbon's **Sheet Metal** tab, choose a command:

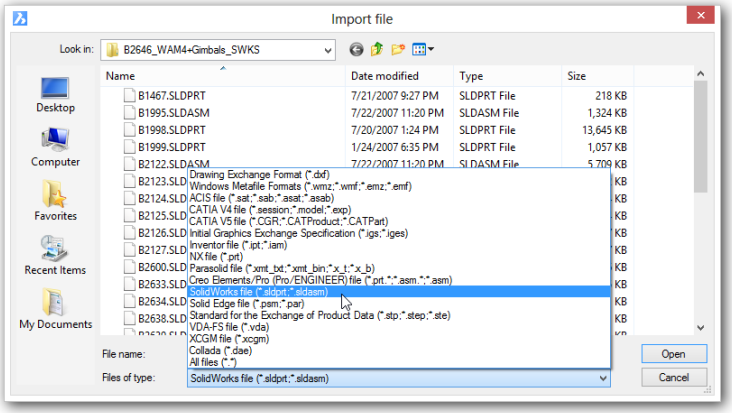


- From the **Sheet Metal** menu, choose a command

BricsCAD Communicator

BricsCAD Communicator is an optional, extra-cost add-on to BricsCAD that provides additional import and export formats. It costs extra, because of the license fees that need to be paid to the firms that write the translators. AutoCAD includes extensive export and import translators at no extra cost through an online service.

BricsCAD Communicator requires BricsCAD Pro or Platinum. When Communicator is installed on your computer, the added file formats appear automatically in the droplists of the Import and Export dialog boxes.



The file types available through the Import dialog box

Import Formats Supported

Standard Formats	AutoCAD	BricsCAD	Description
igs, iges	●	●	Initial Graphics Exchange Specification
jt	●		Siemens Jupiter Technology
x_t, xmt_txt, x_b, xmt_bin	●	●	Siemens Parasolid
ste, stp, step	●	●	Standard for Exchange of Product data
vda		●	VDA-FS
xcgm		●	XML-based CGM
Proprietary Formats			
model, catpart, catproduct	●	●	CATIA V4 and V5 (Windows only)
asm, prt	●	●	Creo Elements / Pro Engineer
iam, ipt	●	●	Inventor
prt	●	●	NX
3dm	●		Rhino
par, psm		●	Solid Edge
sldasm, sldprt	●	●	Solidworks

Export Formats Supported

Standard Formats	AutoCAD	BricsCAD	Description
igs, iges	•	•	Initial Graphics Exchange Specification
ste, stp, step		•	Standard for Exchange of Product data
stl	•	•	Stereolithography
vda		•	VDA-FS
Proprietary Formats			
eps	•		Adobe Encapsulated PostScript
pdf	•	•	Adobe 3D PDF (Windows only)
model, catpart, catproduct		•	CATIA V4 and V5 (Windows only)

TIP When assembly file files are imported, the **ImportProductStructure** variable determines if models are imported as plain geometry or mapped to product structure as native blocks or mechanical components.

Accessing Import and Export Commands

- Enter the **Import** or **Export** command
- In the ribbon's **Home** tab, choose a command from the **File** panel

- From the **File** menu, choose **Import** or **Export**

Import-Export without Communicator

BricsCAD includes 3D import and export translators that are free, independent of Communicator. The 3D file formats supported are as follows:

Import Formats	Description
dwg	AutoCAD drawing file compatible with 2013-2017
dxf	AutoCAD drawing interchange format
dae	Collada (COLLAborative Design Activity)
ifc	Industry foundation classes for BIM
skp	SketchUp

Export Formats	Description
dwg	AutoCAD drawing file compatible with 2013-2017
dxf	AutoCAD drawing interchange format
dwf	Autodesk 3D DWF v6.01
stl	Stereolithography used for 3D printing
dae	Collada (COLLAborative Design Activity)
ifc	Industry foundation classes for BIM

TIP Use the SaveAs command to save to AutoCAD formats older than 2013 — all the way back to Release 14 for DWG and Release 9 for DXF.

As this chapter illustrates, BricsCAD is in many areas of 3D design more capable than AutoCAD. Bricsys is currently targeting BIM and mechanical design, which is why it doesn't offer the industrial design-oriented 3D surfacing commands found in AutoCAD.

APPENDIX A

Command Name Cross-reference

THIS APPENDIX LISTS THE NAMES OF COMMANDS FOUND IN BRICSCAD V17 AND AUTOCAD The list is sorted alphabetically by command name for both CAD packages. When there are no exact matches, notes suggest equivalent command names.

Command names added since the initial V16 edition of this ebook are shown in blue.

Command names specific to the demo, Pro, and Platinum versions of BricsCAD are shown in **boldface**, specifically for BIM modeling (*bim*- commands), mechanical and direct modeling (*bm*- and *dm*- commands), and 3D constraints. These commands are not available in the Standard version. Command names found in the optional, extra-cost sheetmetal (*sm*- commands) add-on are not in this appendix.

This appendix also lists command names removed from recent releases of BricsCAD, along with their replacements, if any. Commands specific to AutoCAD's Block Editor environment are not listed, as BricsCAD does not support it. Not necessarily included are command names that are undocumented by either vendor, nor are names of hardwired aliases or deprecated commands.

AutoCAD Command	BricsCAD Command	Notes
A Commands		
About	About	
AcisIn	AcisIn	
AcisOut	AcisOut	
ActBasepoint	...	
ActManager	...	
ActRecord	...	In BricsCAD, use RecScript
ActStop	...	In BricsCAD, use RecScript
ActUserInput	...	
ActUserMessage	...	
AdCenter, AdcClose	...	In BricsCAD, use Explorer
AdcNavigate	...	
...	AddInMan	VBA COM Add-In Manager for BricsCAD
AddSelected	AddSelected	
Adjust	...	In BricsCAD, use ImageAdjust
Ai_Box	Ai_Box	
Ai_Cone	Ai_Cone	
Ai_Cylinder	Ai_Cylinder	
Ai_Dish	Ai_Dish	
Ai_Dome	Ai_Dome	
...	Ai_EdgeSurf	In AutoCAD, use EdgeSurf
Ai_Mesh	...	In BricsCAD, use Mesh
Ai_Pyramid	Ai_Pyramid	
...	Ai_RevSurf	In AutoCAD, use RevSurf
...	Ai_RuleSurf	In AutoCAD, use RuleSurf
Ai_Sphere	Ai_Sphere	
...	Ai_TabSurf	In AutoCAD, use TabSurf
Ai_Torus	Ai_Torus	
Ai_Wedge	Ai_Wedge	
...	AiMleaderEditAdd	In AutoCAD, use MLeaderEdit
...	AiMleaderEditRemove	In AutoCAD, use MLeaderEdit
Align	Align	
...	AlignSpace	In BricsCAD, aligns viewports
AmeConvert	...	
AnalysisCurvature	...	
AnalysisDraft	...	
AnalysisOptions	...	
AnalysisZebra	...	
AniPath	...	
AnnoReset	AnnoReset	
AnnoUpdate	AnnoUpdate	
Aperture	Aperture	
...	Apparent	In AutoCAD, use -Osnap Apparent
AppAutoLoader	...	
AppLoad	AppLoad	
Arc	Arc	
Archive	...	

AutoCAD Command	BricsCAD Command	Notes
Area	Area	
Array, -Array	Array, - Array	In BricsCAD, now supports dynamic, editable arrays
ArrayClassic	ArrayClassic	
ArrayClose	ArrayClose, -ArrayClose	
ArrayEdit	ArrayEdit	
...	ArrayEditExt	In BricsCAD, edits entities in arrays.
ArrayPath	ArrayPath	
ArrayPolar	ArrayPolar	
ArrayRect	ArrayRect	
Arx	...	In BricsCAD, use AppLoad
Attach	...	In BricsCAD, use ImageAttach, Xref, PdfAdjust
AttachURL	...	In BricsCAD, use Hyperlink
AttDef	AttDef, -AttDef	
AttDisp	AttDisp	
AttEdit	AttEdit	
AttExt	AttExt, -AttExt	
AttIPedit	...	
AttRedef	AttRedef	
AttSync	AttSync	
Audit	Audit	
...	AutoComplete	
AutoConstrain	...	In BricsCAD, use the GcCoincident command's AutoConstrain option
AutoPublish	...	

B Commands

Background	Background	This command is not yet supported in BricsCAD
Base	Base	
BAttMan	BAttMan	
BAttOrder	...	
BEdit	...	In BricsCAD, use Properties to edit dynamic blocks
BESettings	...	Dynamic blocks used in BricsCAD, but not created or edited
BHatch	BHatch, -BHatch	
...	**bimAttachComposition**	Attaches BIM compositions (wall styles) to solids
...	bimAttachSpatialLocation	Locates the drawing in mapping references
...	**bimCheck**	Reports the number of BIM entities in drawings
...	**bimClassify**	Classifies entities as a wall, slab, column, beam, window, or door
...	**bimConnect**	Creates L-connections between two solids
...	**bimDrag**	Extends walls or slabs; modifies their thickness
...	**bimExport**	Exports models to IFC files with all 3D geometric and BIM-related data
...	**bimFlip**	Flips starting faces of compositions; mirrors inserts like windows and doors
...	**bimGetStatisticalData**	Reports statistics data of BIM objects in the current drawing
...	**bimIfcImport**	Imports IFC files
...	**bimInsert**	Inserts window and doors
...	**bimList**	Reports DXF-style data on BIM entities in drawings
	bimPatch	Reserves an of a BIM model for editing with the RefEdit command.
...	**bimReposition**	Repositions inserts (doors, windows) in the faces of solids

AutoCAD Command	BricsCAD Command	Notes
	bimRoom	Defines room areas with markers.
...	**bimSection**	Creates sections from BIM models
...	**bimSectionOpen**	Opens drawing files related to BIM sections
...	**bimSectionUpdate**	Exports BIM sections; also updates BIM sections
...	**bimSkpImport**	Imports SketchUp SKP files
...	**bimSplit**	Automatically separates segmented solids, or by selection of cutting faces
...	**bimUpdateThickness**	Reapplies overall thickness of compositions to solids
...	**bimWindowArray**	Places an array of inserts, such as windows and doors
...	**bimWindowPrint**	Prints windowed areas of models
...	**bimWindowUpdate**	Updates openings when definitions of doors and windows change
Blend	...	
Blipmode	Blipmode	
Blend	...	
Block	Block, -Block	
BlockIcon	...	Required by AutoCAD for old drawings
...	**bmBom**	Inserts bills of material (BOM) tables into drawings
...	**bmBrowser**	Opens and closes the Mechanical Browser bar
...	**bmDependencies**	Lists the names of the files that create assemblies
...	**bmDissolve**	Dissolves mechanical components inserted into drawings
...	**bmExternalize**	Converts local components to external components
...	**bmForm**	Forms new mechanical components and insert them into drawings
...	**bmHardware, -bmHardware**	Inserts standard hardware parts as mechanical components
...	**bmHide**	Hides mechanical components
...	**bmInsert, -bmInsert**	Inserts existing mechanical components into drawings
...	**bmLocalize**	Converts external components to local components
...	**bmMassProp**	Calculates mass properties of components, taking into account density
...	**bmMech**	Converts the current drawing into a mechanical component
...	**bmNew**	Creates a new mechanical component as a new drawing
...	**bmOpen**	Opens parts from assembles for editing
...	**bmOpenCopy**	Creates new drawing with a copy of selected components.
BmpOut	BmpOut	
...	**bmRecover**	Recovers broken mechanical structures
...	**bmReplace**	Replaces component inserts
...	**bmShow**	Shows hidden mechanical components
...	**bmUnMech**	Converts mechanical components into plain drawings
...	**bmUpdate**	Updates the hierarchy of mechanical components
...	**bmVStyle**	Specifies the visual style of components
...	**bmXConvert**	Converts now-obsolete X-Hardware solids to mechanical components
Boundary	Boundary, -Boundary	
Box	Box	
Break	Break	
BRep	...	
Browser	Browser	

C Commands

Cal	Cal	BricsCAD displays Windows Calculator
Camera	Camera	

AutoCAD Command	BricsCAD Command	Notes
...	Center	In AutoCAD, use -Osnap Center
Chamfer	Chamfer	
ChamferEdge	...	In BricsCAD, use dmChamfer
Change	Change	
...	ChapooAccount	In AutoCAD, use Autodesk 360
...	ChapooDownload	Downloads files from Chapoo storage
...	ChapooLogOff	Logs off your Chapoo account
...	ChapooLogOn	Logs into your Chapoo account
...	ChapooOpen	Opens a drawing from Chapoo storage
...	ChapooProject	Opens Chapoo online account in default browser
...	ChapooUpload	Saves the current drawing to Chapoo storage
...	ChapooWeb	Opens the Chapoo Web site in default browser
CheckStandards	...	
ChProp	ChProp	
ChSpace	ChSpace	
Circle	Circle	
ClassicGroup	...	In BricsCAD, use Group
ClassicImage	...	In BricsCAD, use Image
ClassicLayer	...	In BricsCAD, use Layer
ClassicXref	...	In BricsCAD, use Xref
CleanScreenOn / Off	...	
...	CleanUnusedVariables	For developer use in BricsCAD
Clip	...	In BricsCAD, use XClip
Close	Close	
CloseAll	CloseAll	
CloseAllOther	...	
Color	Color, -Color	
CommandLine / Hide	CommandLine / Hide	
...	Commands	In AutoCAD, use the ARX command
Compile	...	Required by AutoCAD only for converting PostScript font files
Cone	Cone	
ConstraintBar	ConstraintBar	
ConstraintSettings	...	In BricsCAD, use Settings
...	ContentBrowserClose / Open	Closes and opens the Content Browser panel
ContentExplorer / Close	...	In BricsCAD, use Explorer
Convert	...	Required by AutoCAD only for old drawings
ConvertCTB	ConvertCTB	
ConvertOldLights	ConvertOldLights	Required for old drawings only
ConvertOldMaterials	ConvertOldMaterials	Required for old drawings only
ConvertPoly	ConvertPoly	
ConvertPStyles	ConvertPStyles	
ConvToMesh	...	
ConvToNurbs	...	
ConvToSolid	...	
ConvToSurface	...	
Copy	Copy	
CopyBase	CopyBase	
CopyClip	CopyClip	

AutoCAD Command	BricsCAD Command	Notes
...	CopyEData	In BricsCAD, copies xdata between entities
CopyHist	CopyHist	
CopyLink	...	
CopyToLayer	...	
...	CPageSetup	In AutoCAD, user PageSetup
CUI	CUI	Executes BricsCAD's Customize command
CuiExport, CuiImport	...	In BricsCAD, use File menu in Customize dialog box
CuiLoad, CuiUnload	CuiLoad, CuiUnload	
Customize	Customize	In AutoCAD, use CUI
CutClip	CutClip	
CvAdd, CvRemove	...	
CvHide, CvShow	...	
CvRebuild	...	
Cylinder	Cylinder	

D Commands

DataExtraction	DataExtraction	
DataLink	...	
DataLinkUpdate	...	
DbConfigure	...	
DbConnect, DbClose	...	
DbList	DbList	dc = dimensional constraint
DcAligned	dcAligned	
DcAngular	dcAngular	
DcConvert	dcConvert	
DcDiameter	dcDiameter	
DcDisplay	dcDisplay	
DcForm	...	
DcHorizontal	dcHorizontal	
DcLinear	dcLinear	
DcRadius	dcRadius	
DcVertical	dcVertical	
...	DdAttE	In AutoCAD, use AttEdit
...	DdEdit	Renamed EditText in AutoCAD 2010
...	DdEModes	BricsCAD uses Settings dialog for entity creation
...	DdFilter	BricsCAD uses DdFilter selection menu
...	DdGrips	BricsCAD uses Settings dialog for grips
DdPtype	DdPtype	BricsCAD uses Settings dialog for points
...	DdSelect	BricsCAD uses Settings dialog for entity selection
...	DdSetVar	BricsCAD uses Settings dialog box
...	DdSTrack	BricsCAD uses Settings dialog for snap tracking
DdVPoint	DdVPoint	
Delay	Delay	
DelConstraint	DelConstraint	
...	DelEData	In BricsCAD, erases xdata from entities
DesignFeedOpen / Close	...	In BricsCAD, use Chapoo
DetachURL	...	In BricsCAD, use Hyperlink

AutoCAD Command	BricsCAD Command	Notes
DgnAdjust	...	BricsCAD does not import DGN files
DgnAttach	...	
DgnClip	...	
DgnExport	...	
DgnImport	...	
DgnLayers	...	
DgnMapping	...	
DigitalSign	...	
DimConstraint	DimConstraint	
...	Dish	In BricsCAD, draws 3D solid dishes
Dist	Dist	
DistantLight	DistantLight	
Divide	Divide	
...	**dmAngle3D**	Applies 3D angle constraints; AutoCAD does not support 3D constraints
...	dmAudit	Checks and fixes 3D models.
...	**dmChamfer**	Chamfers edges
...	**dmConincident3D**	Applies 3D coincident constraints
...	**dmConcentric3D**	Applies 3D concentric constraints
...	**dmConstraint3D**	Super command for applying any kind of 3D constraint
...	**dmDeformCurve**	Deforms by moving or rotating edges to a specified set of target curves
...	**dmDeformMove**	Moves or rotates edges
...	**dmDeformPoint**	Transforms points lying on specified faces
...	**dmDelete**	Erases parts and sub-entities
...	**dmDistance3D**	Applies 3D distance constraints
...	**dmExtrude**	Extrudes planar entities and sub-entities
...	**dmFillet**	Rounds edges
...	**dmFix3D**	Applies 3D fix constraints
...	**dmGroup**	Creates new groups, edits them, and dissolves groups
...	**dmMove**	Moves parts and sub-entities
...	**dmParallel3D**	Applies 3D parallel constraints
...	**dmPerpendicular3D**	Applies 3D perpendicular constraints
...	**dmPushPull**	Pushes and pulls faces and closed contours
...	**dmRadius3D**	Applies 3D radial constraints
...	**dmRepair**	Checks, reports, and optionally fixes errors in 3D solids
...	**dmRevolve**	Revolves planar entities and sub-entities
...	**dmRigidSet3D**	Turns a group of components into a set, like a group
...	**dmRotate**	Rotates entities and sub-entities
...	**dmSelect**	Selects 3D subentities, like edges, faces, protrusions, fillets, and blend networks
...	**dmSelectEdges**	Places faces and solids in a selection set
...	**dmSimplify**	Removes unnecessary edges and vertices, merges seam edges, and so on
...	**dmStitch**	*(Undocumented)* Converts watertight region and surface entities to 3D solids; joins non-watertight surfaces to form a singe surface; converts regions to surfaces
...	**dmTangent3D**	Applies 3D tangency constraints
...	**dmThicken**	Converts surface to 3D solids with specified thicknesses
...	dmTwist	Twists 3D solids by an angle
...	**dmUpdate**	Updates 3D models to satisfy constraints
...	Dome	In BricsCAD, draws 3D solid domes
Donut	Donut	

AutoCAD Command	BricsCAD Command	Notes
DownloadManager	...	
Dragmode	Dragmode	
DrawingRecovery / Hide	...	In BricsCAD, use Recover
DrawOrder	DrawOrder	
...	DrawOrderByLayer	In BricsCAD, controls draw order through layer names
DSettings	DSettings	
...	DText	In AutoCAD, use Text
...	DumpState	For use by BricsCAD developers
DView	DView	
DwfAdjust	...	BricsCAD does not import DWF files
DwfAttach	...	
DwfClip	...	
DwfFormat	...	
DwfLayers	...	
...	DwgCodePage	In AutoCAD, use DwgCodePage system variable
DwgConvert	...	In BricsCAD, use the SaveAs command
DwgProps	DwgProps	
DxbIn	...	Required only for CAD\camera support, now obsolete
DxfIn	DxfIn	
DxfOut	DxfOut	

Dimension Commands

Dim	Dim	
...	Dim1	
DimAligned	DimAligned	
DimAngular	DimAngular	
DimArc	DimArc	
DimBreak	...	
DimBaseline	DimBaseline	
DimCenter	DimCenter	
DimContinue	DimContinue	
DimDiameter	DimDiameter	
DimDisassociate	DimDisassociate	
DimEdit	DimEdit	
DimInspect	...	
DimJogged	...	
DimJogLine	...	
...	DimLeader	In AutoCAD, use Leader
DimLinear	DimLinear	
DimOrdinate	DimOrdinate	
DimOverride	DimOverride	
DimRadius	DimRadius	
DimReassociate	DimReassociate	
DimRegen	DimRegen	
DimRotated	DimRotated	
DimSpace	...	
DimStyle, DimStyle	DimStyle, -DimStyle	

AutoCAD Command	BricsCAD Command	Notes
...	DimStyleSet	Sets the working dimension style
DimTEdit	DimTEdit	

E Commands

EAttEdit	EAttEdit	
EAttExt	...	In BricsCAD, use the DataExtraction command
Edge	...	
EdgeSurf	EdgeSurf	
...	EditEData	In BricsCAD, edits xdata
EditShot	...	
Elev	Elev	
Ellipse	Ellipse	
...	Endpoint	In AutoCAD, use -Osnap Endpoint
Erase	Erase	
eTransmit	eTransmit	
Exchange	...	In BricsCAD, use https://www.bricsys.com/en_INTL/support/
...	ExecuteTool	For use by BricsCAD developers
...	ExpBlocks	In AutoCAD, use the AdCenter command
...	ExpFolders	In AutoCAD, use the AdCenter command
Explode	Explode	
...	Explorer	In AutoCAD, use the AdCenter command
Export	Export	
ExportDWF	...	In BricsCAD, use the DwfOut command
ExportDWFx	...	In BricsCAD, use the Export command
ExportLayout	ExportLayout	
ExportPDF	ExportPDF	
ExportSettings	...	
-ExportToAutocad	...	
...	ExpUcs	
Extend	Extend	
...	Extension	In AutoCAD, use -OSnap Extension
ExternalReferences / Close	...	In BricsCAD, use the Xref command
Extrude	Extrude	

F Commands

FbxExport, FbxImport	...	
Field	Field	
...	FileOpen	Opens files at the command prompt
...	Files	Displays Windows' File Explorer
FilesTab, FileTabClose	...	In BricsCAD, drawing tabs are always open
Fill	Fill	
Fillet	Fillet	
FilletEdge	...	In BricsCAD, use the DmFillet command
Filter	...	In BricsCAD, use the DdFilter command
Find	Find	
FlatShot	FlatShot	

AutoCAD Command	BricsCAD Command	Notes
...	Flatten	In BricsCAD, flattens 3D objects with thickness
Freespot	...	In BricsCAD, use the SpotLight command
Freeweb	...	In BricsCAD, use the WebLight command

G Commands

GcCoincident	GcCoincident	Gc = geometric constraint
GcColLinear	GcColLinear	
GcConcentric	GcConcentric	
GcEqual	GcEqual	
GcFix	GcFix	
GcHorizontal	GcHorizontal	
GcParallel	GcParallel	
GcPerpedicular	GcPerpedicular	
GcSmooth	GcSmooth	
GcSymmetric	GcSymmetric	
GcTangent	GcTangent	
GcVertical	GcVertical	
...	GenerateBoundary	Creates closed polylines from faces of 3D solids
GeographicLocation	GeographicLocation	
GeoLocateMe	...	
GeoMap	...	BricsCAD imports GeoTiff files
GeoMapImage	...	In BricsCAD, use the MapConnect command
GeoMapImageUpdate	...	In BricsCAD, use the MapConnect command
GeoMarkLatLong	...	
GeoMarkMe	...	
GeoMarkPoint	...	
GeoMarkPosition	...	
GeomConstraint	GeomConstraint	
GeoRemove	...	In BricsCAD, use the MapConnect command
GeoReorientMark	...	
GotoUrl	...	In BricsCAD, use the OnWeb command
Gradient	Gradient	
...	GradientBkgOff / On	Turns background gradient off and on
GraphicsConfig	...	In BricsCAD, use the RedSdkInfo command
GraphScr	GraphScr	
Grid	Grid	
Group	Group, -Group	
GroupEdit	...	

H Commands

Hatch	Hatch, -Hatch	
HatchEdit	HatchEdit, HatchEdit	
HatchGenerateBoundary	HatchGenerateBoundary	
HatchSetBoundary	...	
HatchSetOrigin	...	
HatchToBack	HatchToBack	

AutoCAD Command	BricsCAD Command	Notes
Helix	Helix	
Help, ?	Help, ?	
...	HelpSearch	Searches for help topics on the command line
Hide	Hide	In AutoCAD, used for wireframe mode only
HideObjects	HideObjects	
HidePalettes	...	
HlSettings	...	
Hyperlink	Hyperlink, -Hyperlink	
HyperlinkOptions	HyperlinkOptions	

I Commands

Id	Id	
IgesImport / Export	...	In BricsCAD, use the optional Communicator add-on
-Image	Image	
ImageAdjust	ImageAdjust	
ImageAttach	ImageAttach, -ImageAttach	
ImageClip	ImageClip	
...	ImageFrame	In AutoCAD, use the ImageFrame system variable
ImageQuality	ImageQuality	
Import	Import	
ImportSkp	...	In BricsCAD, use the bimSkpImport command
Imprint	...	In BricsCAD, use SolidEdit command's Imprint option
InputSearchOptions	...	
Insert	Insert, -Insert	
...	InsertAligned	Inserts multiple and mirrored blocks
...	Insertion	In AutoCAD, use -OSnap Insertion
InsertObj	InsertObj	
Interfere	Interfere	
Intersect	Intersect	
...	Intersection	In AutoCAD, use -OSnap Intersection
...	InvokeTestApp	Runs BCadTestModuleClient, if loaded
IsoDraft	...	
IsolateObjects	IsolateObjects	
Isoplane	Isoplane	

J Commands

Join	Join	
JpgOut	...	
JustifyText	...	

L Commands

LayCur	LayCur	
LayDel	...	In BricsCAD, use Layer command
Layer, -Layer	Layer, -Layer	BricsCAD uses Explorer for layers
LayerPalette, LayerClose	LayersPanelClose / Open	Closes and opens the Layers panel.

AutoCAD Command	BricsCAD Command	Notes
LayerP	LayerP	
LayerPMode	...	In BricsCAD, use LayerPMode system variable
LayerState	LayerState	BricsCAD uses Explorer for layer states
LayerStateSave	...	
LayFrz	LayFrz	
LayIso	LayIso	
LayLck	LayLck	
LayMch	...	
LayMCur	LayMCur	
LayMrg	...	
LayOff, LayOn	LayOff, LayOn	
-Layout	Layout	
LayoutWizard	...	To be supported in a future release of BricsCAD
LayThw	LayThw	
LayTrans	...	
LayULk	LayULk	
LayUnIso	LayUnIso	
LayVpi	...	In BricsCAD, use the Layer command
LayWalk	...	
Leader	Leader	
Lengthen	Lengthen	
...	LicenseManager	In AutoCAD, click Help | About | Product Information
...	LicEnterKey	Enters BricsCAD license key
...	LicProperties	Displays license information
...	LicPropertiesCommunicator	Licence state of the Communicator module
...	LicPropertiesSheetmetal	Licence state of the sheet metal module
Light	Light	
LightList, LightListClose	LightList	BricsCAD uses Explorer for lights
Limits	Limits	
Line	Line	
Linetype	Linetype, -Linetype	BricsCAD uses Explorer for linetypes
List	List	
LiveSection	LiveSection	
Load	Load	
Loft	**Loft**	
LogFileOn, LogFileOff	LogFileOn, LogFileOff	
...	LookFrom	In AutoCAD, use the NavCube command
LtScale	LtScale	
LWeight	LWeight	BricsCAD uses Settings for lineweights

M Commands

...	Mail	Attaches current drawing to new email message
ManageUploads	...	In BricsCAD, use the ChapooUpload command
...	MapConnect	In AutoCAD, use the GeoMapImage command
Markup, MarkupClose	...	BricsCAD does not support markup files
MassProp	MassProp	
MatBrowserClose / Open	MatBrowserClose / Open	

AutoCAD Command	BricsCAD Command	Notes
MatchCell	...	
...	MatchPerspective	Changes the viewpoint in perspective mode to match a background image
MatchProp	MatchProp	
MatEditorOpen / Close	Materials	In BricsCAD, use Explorer to edit materials
MaterialAssign	...	In BricsCAD, use Layer and Properties to assign materials
MaterialAttach	...	In BricsCAD, use Layer and Properties to assign materials
MaterialMap	MaterialMap	
Materials	Materials	
...	MatLib	Displays the Rendering Materials panel
Measure	Measure	
MeasureGeom	...	In BricsCAD, use the Area, Dist, and MassProp commands
Menu	Menu	
...	MenuLoad, MenuUnload	In AutoCAD, use CuiLoad and CuiUnload
Mesh	Mesh	
MeshCap	...	BricsCAD does not support point-defined surface meshes
MeshCollapse	...	
MeshCrease	...	
MeshExtrude	...	
MeshMerge	...	
MeshOptions	...	
MeshPrimitiveOptions	...	
MeshRefine	...	
MeshSmooth	...	
MeshSmoothLess / More	...	
MeshSpin	...	
MeshSplit	...	
MeshUncrease	...	
...	Midpoint	In AutoCAD, use -OSnap Midpoint
MigrateMaterials	...	Required only for old AutoCAD drawings
MInsert	MInsert	
Mirror	Mirror	
Mirror3d	Mirror3d	
MLeader	MLeader	
MLeaderAlign	...	
MLeaderCollect	...	
MLeaderEdit	MLeaderEdit	
...	MLeaderEditText	Edits all aspects of mleaders
MLeaderStyle	MLeaderStyle	
MlEdit	...	In BricsCAD, use Properties
MLine	MLine	
MlStyle	MLStyle	BricsCAD uses Explorer for multiline styles
Model	...	In BricsCAD, double-click inside the viewport
...	ModelProperties	Opens Settings dialog at Modeler section
...	-ModelProperties	Specifies 3D modeling tolerances at the command prompt
Move	Move	
...	MoveEData	Moves xdata between entities
MRedo	...	In BricsCAD, use Redo multiple times
MSlide	MSlide	

AutoCAD Command	BricsCAD Command	Notes
MSpace	MSpace	
MtEdit	...	In BricsCAD, use Properties
MText	MText, -MText	
Multiple	Multiple	
MView	MView	
MvSetup	MvSetup	

N Commands

NavBar	...	
NavSMotion / Close	...	
NavSWheel	...	
NavVCube	...	In BricsCAD, use the LookFrom command
NCopy	...	
...	Nearest	In AutoCAD, use -Osnap Nearest
NetLoad	NetLoad	
New	New	
NewSheetset	NewSheetset	
NewShot	...	
NewView	...	
...	NewWiz	In BricsCAD, begins new drawings with wizard
...	Node	In AutoCAD, use -OSnap Node
...	None	In AutoCAD, use -OSnap None

O Commands

ObjectScale	ObjectScale, -ObjectScale	
Offset	Offset	
OffsetEdge	...	In BricsCAD, use the SolidEdit Offset command
OleLinks	OleLinks	
OleOpen	OleOpen	
OleScale	...	
OnlineAutocad360	...	In BricsCAD, use the ChapooOpen command
OnlineDocs	...	In BricsCAD, use the ChapooOpen command
OnlineOpenFolder	...	In BricsCAD, use the ChapooDownload command
OnlineOptions	...	
OnlineShare	...	In BricsCAD, perform this function online with Chapoo
OnlineSync	...	
OnlineSyncSettings	...	
OnlineUpload	...	In BricsCAD, use the ChapooUpload command
...	OnWeb	Opens Bricsys.com home page; in AutoCAD, use Browser
Oops	Oops	
Open	Open	
OpenDwfMarkup	...	BricsCAD does not support DWG and markup files
OpenFromCloud	...	In BricsCAD, use the ChapooOpen command
OpenSheetset	OpenSheetset, -OpenSheetset	
Options	Options	
Ortho	Orthogonal	

AutoCAD Command	BricsCAD Command	Notes
-OSnap	OSnap, -OSnap	
OverKill	OverKill, -OverKill	

P Commands

PageSetup	PageSetup	
Pan	Pan, -Pan	
...	Parallel	In AutoCAD, use -OSnap Parallel
Parameters, ParametersClose	...	In BricsCAD, use bmBrowser
...	-Parameters	Creates and edits parameters at the command line
PartiaLoad	...	
-PartialOpen	...	
PasteAsHyperlink	...	
PasteBlock	PasteBlock	
PasteClip	PasteClip	
PasteOrig	PasteOrig	
PasteSpec	PasteSpec	
PcExtractCenterLine	...	
PcExtractCorner	...	
PcExtractSection	...	
PcInWizard	...	
...	PDF	In AutoCAD, use ExternalReferences command
PdfAdjust	PdfAdjust	
PdfAttach	PdfAttach, -PdfAttach	
PdfClip	PdfClip	
PdfLayers	PdfLayers	
...	PdfOptions	Settings for PDF exports
PEdit	PEdit	
...	PEditExt	Edits polylines at the command line
...	Perpendicular	In AutoCAD, use -OSnap Perpendicular
PFace	PFace	
Plan	Plan	
PlaneSurf	...	
PLine	PLine	
Plot	Plot, -Plot	
PlotStamp	...	In BricsCAD, use Print command's Plot Stamp option
PlotStyle	PlotStyle	
PlotterManager	PlotterManager	
PmToggle	...	
PngOut	...	
Point	Point	
PointCloudAttach	...	BricsCAD does not support point clouds
PointCloudColorMap	...	
PointCloudCrop / Uncrop	...	
PointCloudManager / Close	...	
PointCloudSection	...	
PointCloudStylize	...	
PointLight	PointLight	

AutoCAD Command	BricsCAD Command	Notes
Polygon	Polygon	
PolySolid	PolySolid	
PressPull	...	In BricsCAD, use the dmPushpull command
Preview	Preview	
...	Print	In BricsCAD, operates like AutoCAD's Plot command
ProjectGeometry	...	
...	ProfileManager	In AutoCAD, use Profiles tab of Options command
Properties, PropertiesClose	Properties, PropertiesClose	
PSetupIn	PSetupIn, -PSetupIn	
PSpace	PSpace	
PType	...	In BricsCAD, use the DdPtype command
Publish	Publish, -Publish	
PublishToWeb	...	
Purge	Purge, -Purge	
Pyramid	Pyramid	

Q Commands

QDim	...	QDim removed from BricsCAD V14.1.02
QLeader	QLeader	
QNew	QNew	
...	QPrint	In BricsCAD, plots directly without dialog box
QSave	QSave	
QSelect	QSelect	
QText	QText	
QuickCalc, QcClose	...	In BricsCAD, use the Calc command
...	Quadrant	In AutoCAD, use -OSnap Quadrant
QuickCui	...	In BricsCAD, use the Customize command
QuickProperties	...	
Quit	Quit	
QvDrawing, QvDrawingClose	...	In BricsCAD, use Window menu
QvLayout, QvLayoutClose	...	In BricsCAD, use layout tabs or drawing tabs

R Commands

Ray	Ray	
...	ReassocApp	In BricsCAD, reassociates apps with xdata
Recap	...	BricsCAD does not support point clouds
...	RecordRawInput	For developer use in BricsCAD
Recover, RecoverAll	Recover	
...	RecScript	In BricsCAD, begins recording a script file
Rectang	Rectang	
Redefine	Redefine	
Redo	Redo	
Redraw, RedrawAll	Redraw, RedrawAll	
...	RedSdkInfo	In AutoCAD, use GraphicsConfig
RefClose	RefClose	
RefEdit	RefEdit, -RefEdit	

AutoCAD Command	BricsCAD Command	Notes
RefSet	RefSet	
Regen, RegenAll	Regen, RegenAll	
RegenAuto	RegenAuto	
Region	Region	
Reinit	Reinit	
Rename	Rename, -Rename	BricsCAD uses Explorer to rename styles
Render	**Render, -Render**	
RenderCrop	...	
RenderEnvironment / Close	...	To be supported in a future release of BricsCAD
RenderExposure / Close	...	
RenderOnline	...	
RenderPresets / Close	**RenderPresets**	BricsCAD uses Explorer to set rendering presets
RenderWin	...	To be supported in a future release of BricsCAD
RenderWindow / Close	...	
...	**RenderWinClose**	
ResetBlock	ResetBlock	
Resume	Resume	
RevCloud	RevCloud	
Reverse	...	
Revolve	Revolve	
RevSurf	RevSurf	
Ribbon, RibbonClose	Ribbon, RibbonClose	
Rotate	Rotate	
Rotate3D	Rotate3D	
RPref, RPrefClose	...	To be supported in a future release of BricsCAD
RScript	RScript	
...	RtLook	In AutoCAD, use 3dFly; Rt = realtime
...	RtPan	In AutoCAD, use 3dPan
...	RtRot	In AutoCAD, use 3dOrbit
...	RtRotCtr	In AutoCAD, use 3dOrbit
...	RtRotF	In AutoCAD, use 3dOrbit
...	RtRotX	In AutoCAD, use 3dOrbit
...	RtRotY	In AutoCAD, use 3dOrbit
...	RtRotZ	In AutoCAD, use 3dOrbit
...	RtUpDown	In AutoCAD, use 3dSwivel
...	RtWalk	In AutoCAD, use 3dWalk
...	RtZoom	In AutoCAD, use 3dZoom
RuleSurf	RuleSurf	

S Commands

Save	Save	
...	SaveAll	Saves all open drawings
SaveAs	SaveAs	
...	SaveAsR12	Saves drawings in R12 DWG format
SaveImg	...	To be supported in a future release of BricsCAD; for now use Export or MSlide
SaveToCloud	...	In BricsCAD, use the ChappooUpload command
Scale	Scale	

AutoCAD Command	BricsCAD Command	Notes
ScaleListEdit	ScaleListEdit, -ScaleListEdit	
ScaleText	...	
Script	Script	
ScriptCall	...	
...	Scrollbar	Toggles scroll bars
Section	Section	
SectionPlane	SectionPlane	
SectionPlaneJog	...	
SectionPlaneSettings	SectionPlaneSettings	In BricsCAD, use Explorer for section plane settings
SectionPlaneToBlock	SectionPlaneToBlock	
SectionSpinners	....	
...	Security	Determines whether VBA macros may run
SecurityOptions	SecurityOptions	
Seek	...	In BricsCAD, visit sites such as tracepartsonline.com or grabcad.com
Select	Select	
...	SelectAlignedFaces	Selects all faces coplanar with the selected face
...	SelectAlignedSolids	Selects all solids with faces coplanar to the selected face
...	SelectConnectedFaces	Selects all faces connected to the selected face
...	SelectConnectedSolids	Selects all solids whose faces are connected to the selected face
SelectSimilar	SelectSimilar	
...	SelGrips	In AutoCAD, use Ai_SelAll
SetByLayer	...	
SetiDropHandler	...	
...	Settings	In BricsCAD, displays Settings dialog box
...	SettingsSearch	In BricsCAD, searches Settings dialog from the command line
...	SetUCS	In AutoCAD, use UcsMan
SetVar	SetVar	
...	Shade	In AutoCAD, use VsCurrent
-ShadeMode	ShadeMode, -ShadeMode	
Shape	Shape	
Share	...	
Sheetset, SheetsetHide	Sheetset, SheetsetClose	
Shell	Shell	
ShowPalettes	...	
ShowRenderGallery	...	
SigValidate	...	
...	Singleton	In AutoCAD, use SDI system variable
Sketch	Sketch	
Slice	Slice	
Snap	Snap	
SolDraw	...	In BricsCAD, use the ViewBase command
Solid	Solid	
SolidEdit	SolidEdit	
SolProf	SolProf	
SolView	...	In BricsCAD, use the ViewBase command
SpaceTrans	...	
Spell	Spell	
Sphere	Sphere	

AutoCAD Command	BricsCAD Command	Notes
Spline	Spline	
SplinEdit	...	In BricsCAD, use the Properties command
SpotLight	SpotLight	
Standards	...	
...	StatBar	In AutoCAD, use StatBar system variable
Status	Status	
StlOut	StlOut	
...	StopScript	Stops recording to script file
Stretch	Stretch	
Style	Style, -Style	BricsCAD uses Explorer for styles
StylesManager	StylesManager	
Subtract	Subtract	
SunProperties / Close	SunProperties	
...	SupportFolder	Opens C:\Users\<login>\AppData\Roaming\Bricsys\BricsCAD\V17x64\en_US\Support
SurfBlend	...	
SurfExtend	dmExtrude	
SurfExtractCurve	dmMove	
SurfFillet	dmFillet	
SurfNetwork	...	
SurfOffset	...	
SurfPatch	...	
SurfSculpt	...	
SurfTrim, SurfUntrim	...	
...	SvgOptions	In BricsCAD, opens Settings dialog at SVG Export section
Sweep	Sweep	
SysVarMonitor	...	
SysWindows	SysWindows	

T Commands

Table	Table, -Table	
TablEdit	TablEdit	
TableExport	TableExport	
...	TableMod	In BricsCAD, edits cells
TableStyle	TableStyle	
Tablet	Tablet	
TabSurf	TabSurf	
...	Tangent	In AutoCAD, use -OSnap Tangent
TargetPoint	...	
Taskbar	...	
...	TemplateFolder	Opens C:\Users\<login>\AppData\Local\Bricsys\BricsCAD\V17x64\en_US\Templates
...	TestDbUserIo	For developer use in BricsCAD
...	TestDlg	For developer use in BricsCAD
...	TestFatal	For developer use in BricsCAD
Text	Text, -Text	
TextAlign	...	
TextEdit	....	In BricsCAD, use the DdEdit command
TextScr	TextScr	

AutoCAD Command	BricsCAD Command	Notes
TextToFront	TextToFront	
Thicken	...	In BricsCAD, use the DmExtrude command
TifOut	...	
Time	Time	
TInsert	...	
Tolerance	Tolerance	
-Toolbar	Toolbar, -Toolbar	
ToolPalettes / Close	ToolPalettes, ToolPalettesClose	
...	-ToolPanel	Opens tool panels by name at the command bar
Torus	Torus	
TpNavigate	TpNavigate	
...	Trace	Draws wide lines
Transparency	Transparency	
TreeStat	...	
Trim	Trim	
...	TxtExp	Explodes text

U Commands

U	U	
Ucs	Ucs	
UcsIcon	UcsIcon	
UcsMan	...	In BricsCAD, use the SetUcs command
ULayers	...	In BricsCAD, use the Layer command
Undefine	Undefine	
Undo	Undo	
Ungroup	...	In BricsCAD, use the Group command
Union	Union	
UnisolateObjects	UnisolateObjects	
Units	Units, -Units	
UpdateField	UpdateField	
UpdateThumbsNow	...	
...	Url	In AutoCAD, use the Browser command

V Commands

VbaIde	VbaIde	
VbaLoad	VbaLoad, -VbaLoad	
VbaMan	VbaMan	
...	VbaNew	
VbaRun	VbaRun, -VbaRun	
...	VbaSecurity	
VbaStmt	...	
VbaUnload	VbaUnload	
View	View, -View	BricsCAD uses Explorer for views
ViewBase	**ViewBase**	
ViewComponent	...	
ViewDetail	**ViewDetail**	

AutoCAD Command	BricsCAD Command	Notes
ViewDetailStyle	**ViewDetailStyle**	
ViewEdit	**ViewEdit**	
...	ViewExport	Exports drawings from paper space to model space; destroys 3D information
ViewGo	...	
ViewPlay	...	
ViewPlotDetails	...	
ViewProj	**ViewProj**	
ViewRes	ViewRes	
ViewSection	**ViewSection**	
ViewSectionStyle	**ViewSectionStyle**	
ViewSetProj	...	
ViewSymbolSketch / Close	...	
ViewStd	...	In BricsCAD, use the ViewBase command
ViewUpdate	**ViewUpdate**	
VisualStyles / Close	VisualStyles, -VisualStyles	BricsCAD uses VisualStyles in Explorer
VLisp	...	In BricsCAD, use text editor and VLxxx functions
...	VmlOut	Exports drawings in VML format
VpClip	VpClip	
VpLayer	VpLayer	
VpMax / Min	...	
VPoint	VPoint	
VPorts	VPorts, -Vports	
VsCurrent	VsCurrent	
VSlide	VSlide	
VsSave	...	In BricsCAD, use VisualStyles in Explorer
VTOptions	...	

W Commands

WalkFlySettings	...	
WBlock	WBlock, -WBlock	
...	WCascade	Cascades windows
...	WClose	Closes the current window
...	WCloseAll	Closes all windows
WebLight	WebLight	
WebLoad	...	
Wedge	Wedge	
WhoHas	WhoHas	
...	WhTile	Tiles windows horizontally
...	WiArrange	Arranges iconized windows
WipeOut	WipeOut	
WmfIn	...	To be supported in a future release of BricsCAD
WmfOpts	...	To be supported in a future release of BricsCAD
WmfOut	WmfOut	
...	WNext	In AutoCAD, use drawing tabs
...	WorkSets	In BricsCAD, loads named sets of drawings
WorkSpace	WorkSpace	
...	WPrev	In AutoCAD, use drawing tabs

AutoCAD Command	BricsCAD Command	Notes
WsSave	WsSave	
WsSettings	WsSettings	
...	WvTile	In BricsCAD, tiles windows vertically

X Commands

XAttach	XAttach	
XBind	...	To be supported in a future release of BricsCAD
XClip	XClip	
XEdges	XEdges	
XLine	XLine	
XOpen	XOpen	
Xplode	Xplode	
Xref, -XRef	XRef, -XRef	Explorer for external references

Z Command

Zoom	Zoom	

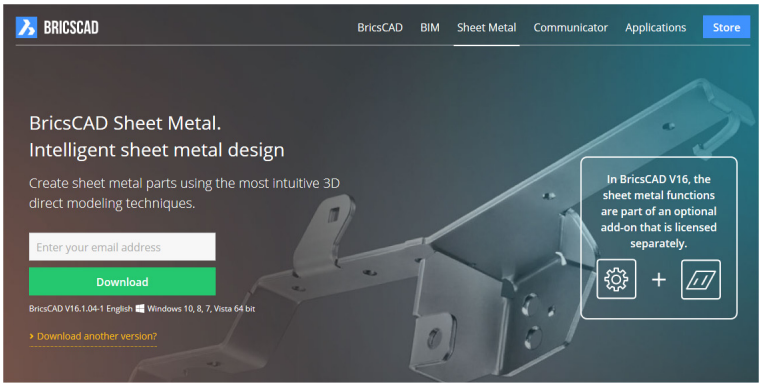
Commands

...	2dIntersection	In AutoCAD, use -OSnap Intersection
3D	3D	
3dAlign	...	
3dArray	3dArray	
3dClip	...	
...	3DCompare	Compares the 3D content of two drawing files
3dCOrbit	...	In BricsCAD, use the RtRot command
...	3dConvert	Converts ACIS solids to polyface meshes
3dDistance		
3dDwf	...	In BricsCAD, use 3D DWF option of Export command
3dEditBar	...	
3dFace	3dFace	
3dFly		In BricsCAD, use the RtLook command
3dFOrbit	...	In BricsCAD, use the RtRot command
...	3dIntersection	In AutoCAD, use -OSnap Intersection
3dMesh	3dMesh	
3dMove	...	In BricsCAD, use Quad cursor's Move option
3dOrbit	...	In BricsCAD, use the RtRot command
3dOrbitCtr	...	
-3dOsnap	...	
3dPan	...	In BricsCAD, use the RtPan command
3dPoly	3dPoly	
3dPrint	...	
3dRotate	...	In BricsCAD, use Quad cursor's Rotate option
3dScale	...	
3dsIn	...	
3dSwivel	...	In BricsCAD, use the RtUpDown command

AutoCAD Command	BricsCAD Command	Notes
3dWalk	...	In BricsCAD, use the RtWalk command
3dZoom	...	In BricsCAD, use the RtZoom command

SHEET METAL MODELING COMMANDS

BricsCAD has the ability to construct sheet metal parts; this function is not available in AutoCAD. These commands are available only in a separate, extra-cost, add-on module to BricsCAD V17. See https://www.bricsys.com/en_INTL/sheetmetal/.



Bricsys Web page for obtaining the Sheet Metal add-on

Blue indicates commands new since the last edition of this ebook:

smBendCreate converts sharp edges between flange faces to bends
SmBendSwitch converts bends to lofted bends
smConvert recognizes flanges and bends in a 3D solids automatically
smDelete removes junctions by restoring sharp edge between two flanges
smDissolve dissolves sheet metal features
smExport2D exports sheet metal as unfolded representation of 2D profiles in DXF or DWG format
smExportOsm export a sheet metal designs in Open Sheet Metal (.osm) format
smFlangeBase creates sheet metal models from closed 2D polylines or regions
smFlangeBend bends existing flanges along a line, taking into account the k-factor
smFlangeConnect closes gaps between two flanges; their orientation does not matter
smFlangeEdge bends the sheet metal to make flanges; generates corner and bend reliefs automatically
smFlangeRotate changes the bend angle of flanges
smFlangeSplit splits flanges along a line drawn on their faces
smForm adds forms to sheet metal
smJunctionCreate converts hard edges into junctions
smJunctionSwitch changing symmetrical junctions to ones with overlapping faces
smLoft constructs sheet metal bodies with lofted bends and flanges
smReliefCreate creates proper corner and bend reliefs.
smRepair joins connected lofted bends surrounded by flanges and rebuilds them tangent to adjacent flanges
smReplace replacing form features with ones from libraries
smRethicken restores 3D solid models from sheet metal part by thickening one side.
smSelectHardEdges selects all hard edges, and then reports about them in the report panel
smUnfold unfolds sheet metal bends

APPENDIX B

System Variable Cross-reference

THIS APPENDIX COMPARES THE NAMES AND VALUES OF VARIABLES FOUND IN AUTOCAD and BricsCAD, listed in alphabetical order. BricsCAD alone has 855 variables.

In addition to supporting many AutoCAD-like system variables, BricsCAD employs *preference variables*, which are unique to it and provide greater access to system settings. The table in this chapter uses the following notations:

- System variables and preference names new since the last edition of this ebook are shown in **blue text**
- BricsCAD preferences are shown in **boldface text**
- Undocumented BricsCAD system variables and preferences are shown in *italicized text*; undocumented AutoCAD ones are not listed

Both CAD programs can change the values of variables, when the variables are not read-only. At the command line, enter the **SetVar** command, and then the name of the system or preference variable. For changing their values through dialog boxes, use these commands:

For **AutoCAD** system variables, enter the name in the **SysVDlg** command

For **BricsCAD** system and preference variables, enter the name in the search field of the **Settings** command

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
A Variables				
AcadLspAsDoc	0	0	AcadLspAsDoc	
AcadPrefix	c:\users\...	C:\Users\...	AcadPrefix	
AcadVer	18.2	20.0 BricsCAD	AcadVer	
...		-1	**AcisHlrResolution**	Hidden-line removal resolution
AcisOutVer	70	70	AcisOutVer	
...		*(not used)*	AcisSaveAsMode	Specifies how to save solids to R12
ActPath	""		...	
ActRecorderState	0		...	
ActRecPath	c:\users\...		...	
ActUi	6		...	
AeCeIpInProgress	off		...	
AFlags	16	0	AFlags	
...		1	AllowTabExternalMove	Allows one tab to be moved to another spot
...		1	AllowTabMove	Allows tabs to be moved horizontally
...		1	AllowTabSplit	Allows tabs to be split
AngBase	0	0	AngBase	
AngDir	0	0	AngDir	
AnnoAllVisible	1	On	AnnoAllVisible	
AnnoAutoScale	-4		...	
AnnoMonitor	-2		...	
...		0	AnnoSelected	
AnnotativeDwg	0	0	AnnotativeDwg	
...		2	**AntiAliasRender**	Level of anti-aliasing in renderings
...		2	**AntiAliasScreen**	Level of anti-aliasing in 3D views
ApBox	0	0	ApBox	
Aperture	10	10	Aperture	
AppAutoLoad	14		...	
AppFrameResources	pack://application...		...	
ApplyGlobalOpacities	0		...	
Area	0	0	Area	
ArrayAssociativity	1	1	ArrayAssociativity	
ArrayEditState	0	0	ArrayEditState	
ArrayType	0		...	
AttDia	0	0	AttDia	
AttIpe	0		...	
AttMode	1	1	AttMode	
AttMulti	1		...	
...		3	**AttractionDistance**	Specifies grips attraction distance
AttReq	1	1	AttReq	
AuditCtl	0	0	AuditCtl	
...		0	AuditErrorCount	Reports number of errors in audit
AUnits	0	0	AUnits	
AuPrec	0	0	AuPrec	
...		0.3	AutoCompleteDelay	Delay before autocomplete appears
...		15	AutoCompleteMode	Determines the autocomplete functions

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
AutoDwfPublish	0		...	
AutomaticPub	0		...	
...		1	AutoMenuLoad	Specifies which menu to load
...		0	AutoResetScales	Deletes unused annotations scales
...		1	**AutosaveChecksOnlyFirstBitDbMod**	Checks first bit only of DbMod for autosave
AutoSnap	63	63	AutoSnap	
...		171	**AutoTrackingVecColor**	Specifies color of the tracking vector
...		1	AutoVpFitting	Fits model to viewport borders automatically
...		*(not used)*	AxisMode	Toggles axis display
...		*(not used)*	AxisUnit	Specifies axis units

B Variables

BackgroundPlot	2	2	BackgroundPlot	
BackZ	0	0	BackZ	
BActionBarMode	1		...	
BActionColor	7		...	
...		""	BaseFile	Specifies default template path & file name
BConStatusMode	0		...	
BDependencyHighlight	1		...	
BGripObjColor	141		...	
BGripObjSize	8		...	
BindType	0	0	BindType	
...		256	BkgColor	Specifies background color
...		256	BkgColorPs	Specifies paper space background color
BlipMode	0	0	Blipmode	
BlockEditLock	0	0	BlockEditLock	
BlockEditor	0	0	BlockEditor	
...		C:\Users\...	**BlocksPath**	Specifies path to blocks for Insert command
...		0	BmReportPanel	
...		1000	BndLimit	
BlockTestWindow	0		...	
BParameterColor	170		...	
BParameterFont	simplex.shx		...	
BParameterSize	12		...	
BpTextHorizontal	1		...	
BtMarkDisplay	1		...	
BvMode	0	0	BvMode	

C Variables

...		1	CacheLayout	Toggles caching of layouts
CacheMaxFiles	256		...	
CacheMaxTotalFiles	1024		...	
CalcInput	1		...	
CameraDisplay	0	0	CameraDisplay	
CameraHeight	0	0	CameraHeight	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
CAnnoScale	1:1	1:1	CAnnoScale	
CAnnoScaleValue	1	1	CAnnoScaleValue	
CaptureThumbnails	1		...	
CBarTransparency	50		...	
CConstraintForm	0		...	
CDate	20090722.2	20090722.15	CDate	
CDynDisplayMode	0		...	
CeColor	bylayer	BYLAYER	CeColor	
CeLtscale	1	1	CeLtScale	
CeLtype	bylayer	BYLAYER	CeLtype	
CeLweight	-1	-1	CeLweight	
CenterMt	0		...	
CeTransparency	ByLayer	ByLayer	CeTransparency	
CGeoCs	""	""	CGeoCs	
ChamferA	0	0.5	ChamferA	
ChamferB	0	0.5	ChamferB	
ChamferC	0	1	ChamferC	
ChamferD	0	0	ChamferD	
ChamMode	0	0	ChamMode	
...		0	**ChapooLog**	Toggles log that records Chapoo activity
...		0	**ChapooLogVerbose**	Toggles added details in Chapoo log
...		1	**ChapooModified**	Action to take on local modified drawings
...		www.mychapoo.com	**ChapooServer**	Reports address of Chapoo server
...		"C:\users\..."	**ChapooTempFolder**	Stores name of local Chapoo folder
...		1	**ChapooUploadDependencies**	Specifies files to upload with drawing
...		www.mychapoo.com	**ChapooWebsite**	Names the Chapoo Web site
...		0	CheckDwlPresence	Checks for DWL drawing lock file
CipMode	0		...	
CircleRad	0	0	CircleRad	
CLayer	0	0	CLayer	
CLayout	"Model"		...	
CleanScreenState	0		...	
...		7	**ClipboardFormat**	Specifies default DWG format for Clipboard
...		127	ClipboardFormats	
...		1	CliState	Reports visibility of command line
CliPromptLines	10		...	
CliPromptUpdate	0		...	
...		0	**CloseChecksOnlyFirstBitDbMod**	Does not save drawing if it was only viewed
CMaterial	bylayer	""	CMaterial	
CmdActive	1	1	CmdActive	
CmdDia	1	1	CmdDia	
CmdEcho	1	1	CmdEcho	
CmdInputHistoryMax	20		...	
...		#f8f8f8	**CmdLineEditBgColor**	Specifies command line background color
...		#000000	**CmdLineEditFgColor**	Specifies command line foreground color
...		Courier New	**CmdLineFontName**	Specifies command line font name
...		10	**CmdLineFontSize**	Specifies command line font size

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		#ffffdd	**CmdLineListBgColor**	Specifies command line background color
...		#000000	**CmdLineListFgColor**	Specifies command line foreground color
...		:	CmdLnText	Specifies prompt prefix
CmdNames	setvar	Options	CmdNames	
CmFadeColor	60		...	
CmFadeOpacity	40		...	
CMleaderStyle	standard	standard	CMleaderStyle	
CMlJust	0	0	CmlJust	
CMlScale	1	1	CmlScale	
CMlStyle	standard	STANDARD	CmlStyle	
CmOsnap	1		....	
ColorTheme	0		...	
...		1	ColorX	Specifies X axis color
...		3	ColorY	Specifies Y axis color
...		5	ColorZ	Specifies Z axis color
...		0	**ComAcadCompatibility**	Checks registry for VB app compatibility
CommandPreview	1		...	
Compass	0	0	Compass	
ComplexLtPreview	1		...	
ConstraintBarDisplay	1		ConstraintBarDisplay	
...		0	ContinuousMotion	Toggles continued motion after release
ConstraintBarMode	4095		...	
ConstraintInfer	0		...	
ConstraintNameFormat	2		...	
ConstraintRelax	0		...	
ConstraintSolveMode	1		...	
ContentExplorerState	0		...	
Coords	1	1	Coords	
CopyMode	0	0	CopyMode	
CPlotStyle	bycolor	ByColor	CPlotStyle	
CProfile	<<unnamed profile>>	DEFAULT	CProfile	
...		1	CreateViewports	Creates viewports in new layouts
CrossingAreaColor	100	3	CrossingAreaColor	
CShadow	casts and receives shadows		...	
CTab	model	Model	CTab	
CTableStyle	standard	STANDARD	CTableStyle	
...		1	Ctrl3DMouse	Toggles use of 3D mouse
...		1	CtrlMouse	Toggles meaning of mouse shortcuts
CullingObj	1		...	
CullingObjSelection	0		...	
CursorBadge	2		...	
CursorSize	5	5	CursorSize	
CVPort	2	2	CvPort	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
D Variables				
DataLinkNotify	2		...	
Date	2455035.85	2455035.63	Date	
DbcState	0	0	DbcState	
DblClkEdit	on	1	DblClkEdit	
DbMod	5	0	DbMod	
DctCust	"c:\users\..."	""	DctCust	
DctMain	enu	en_US.dic	DctMain	
...		2	**ddBetweenKnots**	Distance between knots on NURBS surfaces
...		0	**ddFastMode**	Displays faster with more display errors
...		0	**ddGridAspectRatio**	Specifies the grid aspect ratio
...		0	**ddMaxFacetEdgeLength**	Specifies Maximum edge length of cell sides
...		1000	**ddMaxNumGridLines**	Specifies max grid lines for subdivisions
...		15	**ddNormalTol**	Specifies max deviation between normals
...		0	**ddPointsPerEdge**	Specifies the number of points per edge
...		0	**ddSurfaceTol**	Max distance between facet and true edge
...		1	**ddUseFacetRes**	Toggles use of the FacetRed sysvar
DefaultGizmo	0		...	
DefaultIndex	0		...	
DefaultILghting	1		DefaultILghting	
DefaultLightingType	1		...	
...		8	**DefaultLightShadowBlur**	Default shadow blur
...		*(none)*	**DefaultNewSheetTemplate**	Names .dwg or .dwt as default template
DeflPlStyle	bycolor	ByColor	Deflplstyle	
DefPlStyle	bycolor	ByColor	DefPstyle	
...		1	DeleteTool	
DelObj	1	1	DelObj	
DemandLoad	3	3	DemandLoad	
DesignFeedState	1	...		
DgnFrame	0	2	DgnFrame	
DgnImportMax	10000000		...	
DgnMappingPath	c:\users\...		...	
DgnOsnap	1	1	DgnOsnap	
DiaStat	1	1	DiaStat	
Digitizer	0		...	
DimConstraintIcon	3		...	
DimContinueMode	1		...	
...		0	**DisplaySnapMarkerInAllViews**	Toggles snap markers in all viewports
...		1	**DisplayTooltips**	Displays snap tooltips
...		1	*DispPaperBkg*	Toggles paper space background
...		1	*DispPaperMargins*	Displays paper space margins
DispSilh	0	0	DispSilh	Displays silhouette curves
Distance	0	0	Distance	
DivMeshBoxHeight	3		...	
DivMeshBoxLength	3		...	
DivMeshBoxWidth	3		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
DivMeshConeAxis	8		...	
DivMeshConeBase	3		...	
DivMeshConeHeight	3		...	
DivMeshCylAxis	8		...	
DivMeshCylBase	3		...	
DivMeshCylHeight	3		...	
DivMeshPyrBase	3		...	
DivMeshPyrHeight	3		...	
DivMeshPyrLength	3		...	
DivMeshSphereAxis	12		...	
DivMeshSphereHeight	6		...	
DivMeshTorusPath	8		...	
DivMeshTorusSection	8		...	
DivMeshWedgeBase	3		...	
DivMeshWedgeHeight	3		...	
DivMeshWedgeLength	4		...	
DivMeshWedgeSlope	3		...	
DivMeshWedgeWidth	3		...	
...		1	DmAutoUpdate	Toggles auto update of 3D constrained models
...		0	DmExtrudeMode	Specified operation of Auto mode
...		127	DmRecognize	Determines which 3D constraints are applied
...		1	**DockPriority**	Determines docking priority of toolbars
...		0	**DocTabPosition**	Location of drawing tabs
DonutId	0.5	0.5	DonutId	
DonutOd	1	1	DonutOd	
DragMode	2	2	DragMode	
...		0	DragModeHide	Specifies entities to show while dragging
...		1	DragModeInterrupt	Toggles interrupts of redraws
...		1	DragOpen	Inserts or opens dragged files
DragP1	10	10	DragP1	
DragP2	25	25	DragP2	
...		Off	DragSnap	Controls snap behavior while dragging
DragVs	""		...	
...		C:\Users\...	**DrawingPath**	Additional folders to open drawings
...		"none"	DrawingViewPreset	Presets for the ViewBase command
...		""	DrawingViewPresetScale	Preset annotation scale for ViewBase cmd
DrawOrderCtl	3	3	DrawOrderCtl	
DTextEd	2		...	
DwfFrame	2	2	DwfFrame	
DwfOsnap	1	1	DwfOsnap	
...		2	**DwfVersion**	Specifies export format of DWF files
DwgCheck	1	0	DwgCheck	
DwgCodepage	ansi_1252	ANSI_1252	DwgCodepage	
DwgName	drawing1.dwg	Drawing1.dwg	DwgName	
DwgPrefix	"c:\users\..."	"C:\Users\..."	DwgPrefix	
DwgTitled	0	0	DwgTitled	
DxEval	12	12	DxEval	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
DynConstraintDisplay	1		...	
DynConstraintMode	1	1	DynConstraintMode	
DynDiGrip	31	31	DynDiGrip	
...		142	**DynDimColorHot**	Specifies dynamic dimension hot color
...		142	**DynDimColorHover**	Specifies dynamic dimension hover color
...		1	**DynDimDistance**	Specifies dynamic dimension distance
...		1	**DynDimLineType**	Specifies dynamic dimension line type
DynDiVis	1	1	DynDiVis	
DynInfoTips	1		...	
...		65	**DynInputTransparency**	Specifies dynamic input field transparency
DynMode	-3	2	DynMode	
DynPiCoords	0		...	
DynPiFormat	0		...	
DynPiVis	1		...	
DynPrompt	1		...	
DynTooltips	1		...	

Dimension Variables

DimADec	0	0	DimADec	
DimAlt	off	0	DimAlt	
DimAltD	2	2	DimAltD	
DimAltF	25.4	25.4	DimAltF	
DimAltRnd	0	0	DimAltRnd	
DimAltTd	2	2	DimAltTd	
DimAltTz	0	0	DimAltTz	
DimAltU	2	2	DimAltU	
DimAltZ	0	0	DimAltZ	
DimAnno	0		...	
DimAPost	""	""	DimAPost	
DimArcSym	0	0	DimArcSym	
DimAssoc	2	2	DimAssoc	
DimASz	0.18	0.18	DimASz	
DimAtFit	3	3	DimAtFit	
DimAUnit	0	0	DimAUnit	
DimAZin	0	0	DimAZin	
DimBlk	""	""	DimBlk	
DimBlk1	""	""	DimBlk1	
DimBlk2	""	""	DimBlk2	
DimCen	0.09	0.09	DimCen	
DimClrD	0	0	DimClrD	
DimClrE	0	0	DimClrE	
DimClrT	0	0	DimClrT	
DimDec	4	4	DimDec	
DimDle	0	0	DimDle	
DimDli	0.38	0.38	DimDli	
DimDsep	.	.	DimDsep	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
DimExe	0.18	0.18	DimExe	
DimExo	0.06	0.06	DimExo	
DimFit	3	3	DimFit	
DimFrac	0	0	DimFrac	
DimFxl	1	1	DimFxl	
DimFxLon	off	0	DimFxLon	
DimGap	0.09	0.09	DimGap	
DimJogAng	45	45	DimJogAng	
DimJust	0	0	DimJust	
DimLayer	"use current"		...	
DimLdrBlk	""	""	DimLdrBlk	
DimLfac	1	1	DimLfac	
DimLim	off	0	DimLim	
DimLtEx1	""	""	DimLtEx1	
DimLtEx2	""	""	DimLtEx2	
DimLtype	""	""	DimLtype	
DimLUnit	2	2	DimLUnit	
DimLwD	-2	-1	DimLwD	
DimLwE	-2	-1	DimLwE	
DimPickbox	5		...	
DimPost	""	""	DimPost	
DimRnd	0	0	DimRnd	
DimSah	off	0	DimSah	
DimScale	1	1	DimScale	
DimSd1	off	0	DimSd1	
DimSd2	off	0	DimSd2	
DimSe1	off	0	DimSe1	
DimSe2	off	0	DimSe2	
DimSho	on	on	DimSho	
DimSoxd	off	0	DimSoxd	
DimStyle	standard	STANDARD	DimStyle	
DimTad	0	0	DimTad	
DimTDec	4	4	DimTDec	
DimTFac	1	1	DimTFac	
DimTFill	0	0	DimTFill	
DimTFillClr	0	BYBLOCK	DimTFillClr	
DimTih	on	1	DimTih	
DimTix	off	0	DimTix	
DimTm	0	0	DimTm	
DimTMove	0	0	DimTMove	
DimTofl	off	0	DimTofl	
DimToh	on	1	DimToh	
DimTol	off	0	DimTol	
DimTolj	1	1	DimTolj	
DimTp	0	0	DimTp	
DimTSz	0	0	DimTSz	
DimTVp	0	0	DimTVp	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
DimTxRuler	on		...	
DimTxSty	standard	STANDARD	DimTxSty	
DimTxt	0.18	0.18	DimTxt	
DimTxtDirection	off		...	
DimTzin	0	0	DimTZin	
DimUnit	2	2	DimUnit	
DimUpt	off	0	DimUpt	
DimZin	0	0	DimZin	

E Variables

EdgeMode	0	0	EdgeMode	
Elevation	0	0	Elevation	
...		0	**EnableAttraction**	Enables grips attraction
...		1	**EnableHyperlinkMenu**	Toggles hyperlink menu
...		0	**EnableHyperlinkTooltip**	Toggles hyperlink tooltips
EnterpriseMenu	.		...	
ErHighlight	1		...	
ErrNo	0	0	ErrNo	
Expert	0	0	Expert	
...		0	**ExpInsAlign**	Aligns blocks with selected entity
...		0	**ExpInsAngle**	Default angle for inserted blocks
...		1	**ExpInsFixAngle**	Fixed rotation angle for inserted blocks
...		1	**ExpInsFixScale**	Fixed scale factor for inserted blocks
...		1	**ExpInsScale**	Default scale factor for inserted blocks
ExplMode	1	1	ExplMode	
ExportEplotFormat	2		...	
ExportModelSpace	0	0	ExportModelSpace	
ExportPageSetup	0	0	ExportPageSetup	
ExportPaperSpace	0	0	ExportPaperSpace	
ExpValue	8.8		...	
ExpWhiteBalance	6500		...	
ExtMax	-1e+20,-1e+20,-1e+20	-1e+20,-1e+20,-1e+20	ExtMax	
ExtMin	1e+20,1e+20,1e+20	1e+20,1e+20,1e+20	ExtMin	
ExtNames	1	1	ExtNames	

F Variables

FacetErDevNormal	40		...	
FacetErDevSurface	0		...	
FacetErGridRatio	0		...	
FacetErMaxEdgeLength	0		...	
FacetErMaxGrid	4096		...	
FacetErMeshType	0		...	
FacetErMinUGrid	0		...	
FacetErMinVGrid	0		...	
FacetErPrimitiveMode	1		...	
FacetErSmoothlev	1		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
FacetRatio	0	0	FacetRatio	
FacetRes	0.5	0.5	FacetRes	
...		1	FeatureColors	Colors solid faces by related features
FbxImportLog	1		...	
FieldDisplay	1	1	FieldDisplay	
FieldEval	31	31	FieldEval	
FileDia	1	1	FileDia	
FileTabPreview	1		...	
FileTabState	1		...	
FileTabThumbHover	1		...	
FilletRad	0	0	FilletRad	
FilletRad3d	1.0		...	
FillMode	1	1	FillMode	
FontAlt	simplex.shx	simplex.shx	FontAlt	
FontMap	"c:\users..."	default.fmp	FontMap	
Frame	3	3	Frame	
FrameSelection	1		...	
FrontZ	0	0	FrontZ	
FullOpen	1	1	FullOpen	
FullPlotPath	1		...	

G Variables

GalleryView	1		...	
...		3771	GdiObjects	
...		0	GenerateAssocViews	Associates dimensions in generated views
GeoLatLongFormat	0	1	GeoLatLongFormat	
GeoMapMode	0		...	
GeoMarkerVisibility	1	1	GeoMarkerVisibility	
GeoMarkPositionSize	1		...	
...		1	GetStarted	Toggles the Get Started dialog box
GfAng	0		...	
GfClr1	rgb:000,000,255		...	
GfClr2	rgb:255,255,153		...	
GfClrLum	1		...	
GfClrState	1		...	
GfName	1		...	
GfShift	0		...	
GlobalOpacity	0		...	
...		2	GlSwapMode	Sets swap mode for GL graphics
...		"#d2d2d2"	**GradientColorBottom**	Bottom color of gradient background
...		"#fafafa"	**GradientColorMiddle**	Middle color of gradient background
...		"#ffffff"	**GradientColorTop**	Top color of gradient background
...		0	**GradientMode**	Specifies 0, 2, or 3-color background
...		252	GridAxisColor	Specifies color of grid's axis lines
GridDisplay	2	3	GridDisplay	
GridMajor	5	5	GridMajor	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		253	GridMajorColor	Specifies color of major grid lines
...		254	GridMinorColor	Specifies color of minor grid lines
GridMode	0	0	GridMode	
GridStyle	0	1	GridStyle	
GridUnit	0.5000,0.5000	10,10,10	GridUnit	
...		1	GridXyzTint	Toggles coloring of x,y,z grid lines
GripBlock	0	0	GripBlock	
GripColor	150	160	GripColor	
GripDynColor	140	140	GripDynColor	
GripHot	12	240	GripHot	
GripHover	11	150	GripHover	
GripMultifunctional	3		...	
GripObjLimit	100	100	GripObjLimit	
Grips	1	1	Grips	
GripSize	5	5	GripSize	
GripSubobjMode	1		...	
GripTips	1	1	GripTips	
GroupDisplayMode	2		...	
...		0	GsDeviceType2D	Selects graphics system for wirefreames
...		0	GsDeviceType3D	Specifies graphics system for hidden, etc.
GtAuto	1		...	
GtDefault	0		...	
GtLocation	1		...	

H Variables

HaloGap	0	0	HaloGap	
Handles	1	1	Handles	
HatchBoundSet	0		...	
HatchType	0		...	
HelpPrefix	"C:\Program..."		...	
HidePrecision	0	0	HidePrecision	
HideText	on	1	HideText	
HideXrefScales	1	1	HideXrefScales	
Highlight	1	1	Highlight	
...		142	HighlightColor	Specifies highlight color
...		0	HighlightEffect	Specifies color use for highlighting
HighlightSmoothing	1		...	
...		1	**HorizonBkg_Enable**	Toggles horizon in perspective views
...		"#878787"	**HorizonBkg_GroundHorizon**	Color of ground at horizon
...		"#5F5F5F"	**HorizonBkg_GroundOrigin**	Color of the ground
...		"#239BFF"	**HorizonBkg_SkyHigh**	Color of the sky at high elevation
...		"#FFFFFF"	**HorizonBkg_SkyHorizon**	Color of sky at horizon
...		"#FAFAFF"	**HorizonBkg_SkyLow**	Color of the sky at low elevation
HpAng	0	0	HpAng	
HpAnnotative	0	0	HpAnnotative	
HpAssoc	1	1	HpAssoc	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
HpBackgroundColor	"."		...	
HpBound	1	1	HpBound	
HpBoundRetain	0		...	
HpColor	"."		...	
HpDlgMode	2		...	
HpDouble	0	0	HpDouble	
HpDrawOrder	3	3	HpDraworder	
HpGapTol	0	0	HpGapTol	
HpInherit	0		...	
HpIslandDetection	1		...	
HpIslandDetectionMode	1		...	
HpLayer	"Use Current"		...	
HpLinetype	Off		...	
HpMaxArea	100		...	
HpMaxLines	1000000		...	
HpName	ansi31	ANSI31	HpName	
HpObjWarning	10000	10000	HpObjWarning	
HpOrigin	0.0000,0.0000	0,0	HpOrigin	
HpOriginMode	0		...	
HpPickMode	0		...	
HpQuickPreview	On		...	
HpQuickPreviewTimeout	2		...	
HpScale	1	1	HpScale	
HpSeparate	0	0	HpSeparate	
HpSpace	1	1	HpSpace	
...		0	HpStyle	Determines hatching of islands
HpTransparency	"."	"."	HpTransparency	
HyperlinkBase	.	.	HyperlinkBase	

I Variables

IBEnvironment	0		...	
...		C:\Users\<login>\...	**ImageCacheFolder**	Path to folder storing image cache files
...		160	**ImageCacheMaxMemory**	Maximum RAM to reserve for image cache
...		1	**ImageDiskCache**	Toggles use of the disk cache for images
ImageFrame	1	1	ImageFrame	
ImageHlt	0	0	ImageHlt	
...		0	ImageNotify	Alert for missing raster attachments
...		0	**ImportCuiFileExists**	Prompt, overwrite, or rename imported CUI
...		1	**IncludePlotStamp**	Toggles plot stamp on plots
Impliedface	1		...	
IndexCtl	0	0	IndexCtl	
InetLocation	www.autodesk.com	www.bricsys.com	InetLocation	
InputHistoryMode	15		...	
InputSearchDelay	300		...	
InsBase	0.0,0.0,0.0	0;0;0	InsBase	
InsName	.	.	InsName	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
InsUnits	1	1	InsUnits	
InsUnitsdefSource	1	1	InsUnitsdefSource	
InsUnitsdefTarget	1	1	InsUnitsdefTarget	
IntelligentUpdate	20		...	
InterfereColor	1	"ByLayer"	InterfereColor	
InterfereObjVs	realistic	""	InterfereObjVs	
InterfereVpVs	3d wireframe	""	InterfereVpVs	
IntersectionColor	257		IntersectionColor	
IntersectionDisplay	off	0	IntersectionDisplay	
ISaveBak	1	1	ISaveBak	
ISavePercent	50	50	ISavePercent	
Isolines	4	4	Isolines	

L Variables

LargeObjectSupport	0		...	
LastAngle	0	0	LastAngle	
LastPoint	5.7,13.5,0.0	0;0;0	LastPoint	
LastPrompt	lastangle	: options	LastPrompt	
Latitude	37.8	37.7950	Latitude	
LayerDlgMode	1		...	
LayerEval	0		...	
LayerEvalCtl	1		...	
LayerFilterAlert	2		...	
LayerNotify	0		...	
...		1	LayerPMode	Toggles tracking of layer changes
LayLockFadeCtl	50	50	LayLockFadeCtl	Amount of fading of locked layers
LayoutCreateViewport	1		...	
LayoutRegenCtl	2	2	LayoutRegenCtl	
LayoutTab	1		...	
LegacyCodeSearch	off		...	
LegacyCtrlPick	0		...	
LensLength	50	50	LensLength	
...		31	LicExpDays	Number of day at which license expires
...		0	LicFlags	Specifies if components are licensed
...		""	LicKey	Reports software license number
...		30	**LightGlyphColor**	Specifies color of light glyphs (icons)
LightGlyphDisplay	1	1	LightGlyphDisplay	
LightingUnits	2	0	LightingUnits	
LightsInBlocks	1		...	
...		1	**LightWebGlyphColor**	Specifies color of glyphs of web lights
LimCheck	0	0	LimCheck	
LimMax	12.0000,9.0000	12;9	LimMax	
LimMin	0.0000,0.0000	0;0	LimMin	
LinearBrightness	0		...	
LinearContrast	0		...	
LineFading	on		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
LineFadingLevel	2		...	
...		1	LispInit	Preserves LISP functions between sessions
Locale	enu	enu	Locale	
...		c:\users\...	**LocalRootFolder**	Specifies path to local root folder
LocalRootPrefix	c:\users\...	c:\users\...	LocalRootPrefix	
LockUi	0		...	
LoftAng1	90	1.5708	LoftAng1	
LoftAng2	90	1.5708	LoftAng2	
LoftMag1	0	0	LoftMag1	
LoftMag2	0	0	LoftMag2	
LoftNormals	1	1	LoftNormals	
LoftParam	7	7	LoftParam	
LogExpBrightness	65		...	
LogExpContrast	50		...	
LogExpDaylight	2		...	
LogExpMidtones	1		...	
LogExpPhysicalScale	1500		...	
LogFileMode	0	0	LogFileMode	
LogFileName	"c:\users\..."	""	LogFileName	
LogFilePath	"c:\users\..."	"c:\users\..."	LogFilePath	
LogInName	<login>	BricsCAD user	LogInName	
Longitude	-122.39	-122.3940	Longitude	
...		1	**LookFromDirectionMode**	Specifies number of LookFrom directions
...		1	**LookFromFeedback**	LookFrom help in tooltips or on status bar
...		1	**LookFromZoomExtents**	Zoom to extents with each LookFrom pick
LtScale	1	1	LtScale	
LUnits	2	2	LUnits	
LuPrec	4	4	LuPrec	
LwDefault	211	25	LwDefault	
LwDisplay	off	0	LwDisplay	
...		0.55	LwDispScale	Specifies lineweight display scale
LwUnits	1	1	LwUnits	

M Variables

...		0	MacroRec	Reports that macro is being recorded
MacroTrace	0	0	MacroTrace	
...		0.01	**MassPropAccuracy**	Specifies accuracy for mass properties
...		"oz lbs stone mg g kg tonne" MassUnits		Units for displaying mass of objects
MaxBrowserState	0		...	
MaterialsPath	""		...	
MaxActVp	64	64	MaxActVp	
MaxHatch	100000	100000	MaxHatch	
MaxSort	1000	1000	MaxSort	
MaxTouches	0		...	
...		0	MaxThreads	Specifies max threads for redraw, regen, loads
MButtonPan	1	1	MButtonPan	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
MeasureInit	0	1	MeasureInit	
Measurement	0	1	Measurement	
MenuBar	0		...	
MenuCtl	1	1	MenuCtl	
MenuEcho	0	0	MenuEcho	
MenuName	"c:\users\..."	"default"	MenuName	
MeshType	1		MeshType	
...		1	**MiddleClickClose**	Closes tabs with middle-button click
...		732374555	MilliSecs	Reports milliseconds since BricsCAD started
MirrHatch	0		...	
MirrText	0	0	MirrText	
MLeaderScale	1	1	MLeaderScale	
ModeMacro	""	""	ModeMacro	
MsLtScale	1	1	MsLtScale	
MsOleScale	1	1	MsOleScale	
MTextAutoStack	1		...	
MTextColumn	2	0	MTextColumn	
MTextDetectSpace	1		...	
MTextEd	internal	Internal	MTextEd	
MTextFixed	2	2	MTextFixed	
MTextToolbar	2		...	
...		0	MtFlags	Controls multi-core redraws, loads, regens
MTJigString	abc		...	
MyDocumentsPrefix	"c:\users\..."		...	

N Variables

NavBarDisplay	1		...	
NavsWheelMode	2		...	
NavsWheelOpacityBig	50		...	
NavsWheelOpacityMini	50		...	
NavsWheelSizeBig	1		...	
NavsWheelSizeMini	1		...	
NavVCubeDisplay	1	1	NavVCubeDisplay	
NavVCubeLocation	0	0	NavVCubeLocation	
NavVCubeOpacity	50	50	NavVCubeOpacity	
NavVCubeOrient	1	1	NavVCubeOrient	
NavVCubeSize	4	4	NavVCubeSize	
NewTabMode	1		...	
...		4	NFileList	Specifies length of recent file list
NoMutt	0	0	NoMutt	
NorthDirection	0	0	NorthDirection	

O Variables

ObjectIsolationMode	0	0	ObjectIsolationMode	
ObscuredColor	257	257	ObscuredColor	
ObscuredLtype	0	0	ObscuredLtype	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
OffsetDist	-1	1	OffsetDist	
...		0	OffsetErase	Determines if source entities are erased
OffsetGapType	0	0	OffsetGapType	
OleFrame	2	2	OleFrame	
OleHide	0	0	OleHide	
OleQuality	3	3	OleQuality	
OleStartup	0	0	OleStartup	
OnlineDocMode	1		...	
OnlineSyncTime	300		...	
OpenPartial	1	1	OpmState	
OrbitAutoTarget	1		...	
OrthoMode	0	0	OrthoMode	
OsMode	4133	4133	OsMode	
OsnapCoord	2	2	OsnapCoord	
OsnapHatch	0		...	
OsnapOverride	0		...	
OsnapZ	0	0	OsnapZ	
OsOptions	3	1	OsOptions	

P Variables

PaletteOpaque	2		...	
...		1	**PanBuffer**	Buffers pans
PaperUpdate	0	0	PaperUpdate	
ParameterCopyMode	1	1	ParameterCopyMode	
ParameterStatus	0		...	
PcmState	0		...	
...		1	**PdfEmbeddedTtf**	Embeds fonts in PDF output
...		2	**PdfExportSolidHatchType**	Min resolution of solid hatches saved to PDF
PdfFrame	1	1	PdfFrame	
...		3000	**PdfHatchToBmpDpi**	Resolution of hatches exported to PDF
...		1	**PdfImageAntiAlias**	Anti-aliases images being upscaled.
...		1	**PdfImageCompression**	Specifies compression for images.
...		300	**PdfImageDPI**	Minimum resolution of images saved to PDF
...		1	**PdfLayersSetting**	Includes layers in PDF files
...		0	**PdfLayoutsToExport**	Exports content of all layouts
...		0	PdfMergeControl	Specifies the look of overlapping lines
...		0	PdfNotify	Alert for missing PDF attachments
PdfOsnap	1	1	PdfOsnap	
...		297	**PdfPaperHeight**	Overrides paper height in PDF files
...		0	**PdfPaperSizeOverride**	Overrides paper size in PDF files
...		210	**PdfPaperWidth**	Overrides paper width in PDF files
...		300	**PdfRenderDPI**	Minimum resolution of renders saved to PDF
...		0	**PdfShxTextAsGeometry**	Exports SHX text as geometry
...		1	**PdfSimpleGeomOptimization**	Optimizes geometry in PDF files
...		0	**PdfTtfTextAsGeometry**	Exports TTF text as geometry
...		1	**PdfUsePlotStyles**	Uses plot styles when plotting to PDF

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		1	**PdfZoomToExtentsMode**	Zooms to extents mode in PDF files
PdMode	0	0	PdMode	
PdSize	0	0	PdSize	
PeditAccept	0	0	PEditAccept	
PEllipse	0	0	PEllipse	
Perimeter	0	0	Perimeter	
Perspective	0	0	Perspective	
PerspectiveClip	5		...	
PfacevMax	4	4	PFaceVMax	
PickAdd	1	1	PickAdd	
PickAuto	1	1	PickAuto	
PickBox	3	3	PickBox	
PickDrag	0	0	PickDrag	
PickFirst	1	1	PickFirst	
PickStyle	0	1	PickStyle	
...		1	**PictureExportScale**	Specifies scale factor for raster exports
...		0	**PictureFolder1**	Sets folder for storing raster images
...		1	**PictureFolder2**	Sets folder for storing raster images
...		3	**PictureFolder3**	Sets folder for storing raster images
...		5	**PictureFolder4**	Sets folder for storing raster images
_PkSer	""	""	_PkSer	
Platform	*varies*	*varies*	Platform	
...		0	PLineCache	Creates a cache of polyline vertices
PlineConvertMode	0	0	PLineConvertMode	
PlineGen	0	0	PLineGen	
PlineType	2	2	PLineType	
PlineWid	0	0	PLineWid	
...		c:\users\...	**PlotCfgPath**	Specifies plotter configuration path
...		""	PlotId	Deprecated; included for compatibility
PlotOffset	0		...	
...		c:\program files...	**PlotOutputPath**	Specifies path to plot output folder
PlotRotMode	2	2	PlotRotMode	
PlotTransparencyMode	1		...	
...		c:\users\...	**PlotStylePath**	Specifies path to plot styles
...		0	Plotter	Specifies path to plotter cfg folder
...		1	PlotterTransparencyOverride	Overrides setting in Print dialog box
PlQuiet	0	0	PlQuiet	
PointCloud2dDisplay	0		...	
PointCloudAutoUpdate	1		...	
PointCloudBoundary	1		...	
PointCloudCacheSize	512		...	
PointCloudDensity	15		...	
PointCloudLighting	2		...	
PointCloudLightSource	0		...	
PointCloudLock	0		...	
PointCloudLod	10		...	
PointCloudPointMax	1500000		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
PointCloudPointMaxLegacy	1500000		...	
PointCloudPointSize	2		...	
PointCloudRtDensity	5		...	
PointCloudShading	0		...	
PointCloudVizRetain	1		...	
PolarAddAng	.	.	PolarAddAng	
PolarAng	90	90	PolarAng	
PolarDist	0	0	PolarDist	
PolarMode	0	0	PolarMode	
PolySides	4	4	PolySides	
Popups	1	1	Popups	
PreviewCreationTransparency	60		...	
...		30	**PreviewDelay**	Delays subentity highlighting under cursor
PreviewEffect		2	PreviewEffect	(Not yet supported)
PreviewFilter	7	7	PreviewFilter	
PreviewType	0	0	PreviewType	
...		1	**PreviewWndInOpenDlg**	Displays preview window in Open dialog box
...		"."	**PrintFile**	Specifies alternative name for print files
...		BricsCAD	Product	Reports the product name
...		1	ProgBar	Toggles progress bar
...		BricsCAD	Program	Reports the product name
ProjectName	.	.	ProjectName	
...		""	**ProjectSearchPaths**	Specifies project names & search paths
ProjMode	1	1	ProjMode	
PropObjLimit	25000		...	
PropertyPreview	1		...	
PropPrevTimeout	1		...	
...		3	PromptMenu	Toggles prompt menu
...		0	**PromptMenuFlags**	Toggles hidden prompts
...		1	PromptOptionTranslateKeywords	Toggles use of international commands
ProxyGraphics	1	1	ProxyGraphics	
ProxyNotice	1	1	ProxyNotice	
ProxyShow	1	1	ProxyShow	
ProxyWebSearch	0	1	ProxyWebSearch	
PsLtScale	1	1	PsLtScale	
PsolHeight	4	80	PSolHeight	
PsolWidth	0.25	5	PSolWidth	
PsProlog	.	""	PsProlog	
PsQuality	75	75	PsQuality	
PStyleMode	1	1	PStyleMode	
PStylePolicy	1	1	PStylePolicy	
PsVpScale	0	0	PsVpScale	
PublishAllSheets	1	1	PublishAllSheets	
PublishCollate	1		...	
PublishHatch	1		...	
PUcsBase	.	.	PUcsBase	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
Q Variables				
QpLocation	0		...	
QpMode	1		...	
QaFlags	0	0	QaFlags	
QtextMode	0	0	QtextMode	
...		20	QuadAperture	Area to search for entities, in pixels
...		1	QuadCommandLaunch	If Quad launches with application
...		0	QuadCommandSort	Specifies sort order of commands
...		1	QuadDisplay	Toggles display of the Quad cursor
...		110	**QuadExpandDelay**	Delay before expanding, in msec
...		50	QuadExpandTabDelay	Delay before expanding underlaying buttons
...		0	**QuadExpandGroup**	Specifies how groups expand
...		0	**QuadGoTransparent**	Toggles Quad's transparent
...		1000	**QuadHideDelay**	Quad cursor display delay after mouse movement
...		40	**QuadHideMargin**	Delay before Quad is hidden, in msecs
...		16	**QuadIconSize**	Toggles between large and small icon
...		1	**QuadIconSpace**	Specifies spacing between icons
...		1	**QuadPopupCorner**	Location of Quad relative to cursor
...		500	**QuadShowDelay**	Quad display delay after entity highlight
...		1200	**QuadTooltipDelay**	Delay before tooltips appear, in msec
...		4	**QuadWarpPointer**	How Quad interacts with cursor
...		5	QuadWidth	Specifies width of Quad, in columns
QvDrawingPin	0		...	
QvLayoutPin	0		...	
R Variables				
RasterDpi	300		...	
RasterPercent	20		...	
RasterPreview	1	1	RasterPreview	
RasterThreshold	20	...		
...		5	**RealtimeSpeedup**	Skips messages during realtime pan
...		1	RealWorldScale	Renders materials at real-world scale factor
Rebuild2dCv	6		...	
Rebuild2dDegree	3		...	
Rebuild2dOption	1		...	
RebuildDegreeU	3		...	
RebuildDegreeV	3		...	
RebuildOptions	1		...	
RebuildU	6		...	
RebuildV	6		...	
...		C:\Users\...	**RecentPath**	Specifies recently used path
RecoveryAuto	0		...	
RecoveryMode	2		...	
...		50	**RedHiliteFull_Edge_Alpha**	Transparency of hidden edges
...		#FFFFFF	**RedHilite_HiddenEdge_Color**	Color of hidden edges

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		1	**RedHilite_HiddenEdge_Smoothing**	
...		1	**RedHilite_HiddenEdge_Thickness**	
...		100	**RedHiliteFull_Edge_Alpha**	Transparency of edges
...		#007AFF	**RedHiliteFull_Edge_Color**	Color of edges
...		0	**RedHiliteFull_Edge_ShowHidden**	Toggle visibility of hidden edges
...		1	**RedHiliteFull_Edge_Smoothing**	Toogle smoothness of edges
...		2	**RedHiliteFull_Edge_Thickness**	Thickness of edges, in pixels
...		10	**RedHiliteFull_Face_Alpha**	Transparency of faces
...		#007AFF	**RedHiliteFull_Face_Color**	Color of faces
...		100	**RedHilitePartial_SelectedEdge_Alpha**	
...		#007AFF	**RedHilitePartial_SelectedEdge_Color**	
...		1	**RedHilitePartial_SelectedEdge_ShowGlow**	
...		1	**RedHilitePartial_SelectedEdge_Smoothing**	
...		2	**RedHilitePartial_SelectedEdge_Thickness**	
...		75	**RedHilitePartial_SelectedEdgeGlow_Alpha**	
...		#FFFFFF	**RedHilitePartial_SelectedEdgeGlow_Color**	
...		1	**RedHilitePartial_SelectedEdgeGlow_Smoothing**	
...		3	**RedHilitePartial_SelectedEdgeGlow_Thickness**	
...		10	**RedHilitePartial_SelectedFace_Alpha**	Transparency of selected faces
...		#007AFF	**RedHilitePartial_SelectedFace_Color**	Color of selected faces
...		1	**RedHilitePartial_UnselectedEdge_ShowHidden**	
...		0	***RefEditLockNotInWorkset***	Locks entities not being edited by RefEdit
RefEditName	""	""	RefEditName	
RegenMode	1	1	RegenMode	
RememberFolders	1	1	RememberFolders	
RenderEnvState	0		...	
RenderLevel	5		...	
RenderLightCalc	1		...	
RenderTarget	0		...	
RenderTime	10		...	
...		C:\ProgramData\...	**RenderMaterialPath**	Path to folder with materials
...		C:\Program Files\...	**RenderMaterialStaticPath**	Path to folder with read-only materials
RenderQuality	1		...	
RenderUserLights	1		...	
...		1	**RenderUsingHardware**	Toggles use of hardware for rendering
ReportError	1		...	
...		0	**RevCloudArcStyle**	Specifies revision cloud arc style
RevCloudCreateMode	1		...	
RevCloudGrips	on		...	
...		0.38	**RevCloudMaxArcLength**	Specifies revision cloud max arc length
...		0.38	**RevCloudMinArcLength**	Specifies revision cloud min arc length
RibbonContextSelLim	2500		...	
RibbonDockedHeight	0	120	RibbonDockedHeight	
RibbonIconResize	1		...	
RibbonSelectMode	1		...	
RibbonState	1	0	RibbonState	
...		c:\users...	**RoamableRootFolder**	Path to user's Roaming folder

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
RoamableRootPrefix	"c:\users\..."	c:\users\...	RoamableRootPrefix	
RolloverOpacity	0		...	
RolloverTips	1		...	
RtDisplay	1	1	RtDisplay	
...		1	RtRotationSpeedFactor	Specifies turning speed
...		1	RtWalkSpeedFactor	Specifies walking speed
...		2	**RunAsLevel**	License level: 0=Classic, 1=Pro, 2=Platinum

S Variables

SafeMode	0		...	
...		1	**SaveChangeToLayout**	Saves print changes to layout
SaveFidelity	1	1	SaveFidelity	
SaveFile	"c:\users\..."	""	SaveFile	
SaveFilePath	"c:\users\..."	C":\Users\..."	SaveFilePath	
...		1	**SaveFormat**	Sets the DWG file format
SaveName	Drawing1.dwg	""	SaveName	
...		1	SaveRoundTrip	Saves entities to preserve them
SaveTime	10	0	SaveTime	
ScreenBoxes	0	26	ScreenBoxes	
ScreenMode	3	1	ScreenMode	
ScreenSize	1366.0,499.0	784.0;506.0	ScreenSize	
...		256	ScrlHist	Specifies number of lines saved in history
...		0	Sdi	Toggles single-document interface
SectionOffsetInc	6.0		...	
SectionThicknessInc	1.0		...	
SecureLoad	1		...	
SelectionAnnoDisplay	1	1	SelectionAnnoDisplay	
SelectionArea	1	1	SelectionArea	
SelectionAreaOpacity	25	25	SelectionAreaOpacity	
SelectionCycling	0		...	
SelectionEffectColor	0		...	
...		0	SelectionModes	Subentities or boundaries to highlight
SelectionPreview	3	3	SelectionPreview	
SelectionPreviewLimit	2000		...	
SelectSimilarMode	130	130	SelectSimilarMode	
SetByLayerMode	127		...	
ShadEdge	3	3	ShadEdge	
ShadeDif	70	70	ShadeDif	
ShadowPlaneLocation	0		...	
...		1	**SheetNumberLeadingZeroes**	Number of zeros to prefix sheet numbers
...		1	**SheetSetAutoBackup**	Makes backups of sheet files
...		C:\Users\...	**SheetSetTemplatePath**	Path to the sheetset templates folder
ShortcutMenu	11	2	ShortcutMenu	
ShortcutMenuDration	250		...	
...		1	**ShowDocTabs**	Toggles drawing tabs on
...		0	**ShowFullPathInTitle**	Displays full path in title bar

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
ShowHist	1		...	
ShowLayerUsage	0	0	ShowLayerUsage	
ShowmotionPin	1		...	
ShowPageSetupForNewLayouts	0	...		
...		1	**ShowScrollButtons**	Toggles display of scroll buttons
...		0	**ShowTabCloseButton**	Toggles display of Close button on tabs
...		0	**ShowTabCloseButtonActive**	Toggles display of Close button on active tab
...		1	**ShowTabCloseButtonAll**	Toggles display of Close button on all tabs
...		1	**ShowTabControls**	Toggles display of tabs
...		1	**ShowWindowListButton**	Toggles display of droplists
ShpName	""	""	ShpName	
SigWarn	1		...	
...		0	**SingletonMode**	Toggles multiple BricsCAD instances
SketchInc	0.1	0.1	SketchInc	
SkPoly	0	0	SkPoly	
SkTolerance	0.5		...	
SkyStatus	0		SkyStatus	
...		#FFDC50	SmColorBend	Color of sheet metal bends
...		#64D296	SmColorBendRelief	Color of sheet metal reliefs
...		#64D296	smColorCornerRelief	Color of sheet metal corners
...		#90A4AE	SmColorFlange	Color of sheet metal flanges
...		#FF6E40	SmColorJunction	Color of sheet metal junctions
...		#A0DCFA	SmColorLoftedBend	Color of sheet metal lofted bends
...		3	SmLayerColorAnnotations	Layer color of unfolded dimensions
...		5	SmLayerColorBendAnnotations	Layer color of unfolded annotations
...		1	SmLayerColorBendLine	Layer color of unfolded bend lines
...		7	SmLayerColorContours	Layer color of unfolded 2D DXF output
...		""	SmTargetCAM	Specifies the intended CAM system
SmoothMeshConvert	0		...	
SmoothMeshGrid	3		...	
SmoothMeshMaxFace	838300		...	
SmoothMeshMaxLev	4		...	
SnapAng	0	0	SnapAng	
SnapBase	0.0000,0.0000	0;0	SnapBase	
SnapGridLegacy	0		...	
SnapIsoPair	0	0	SnapIsoPair	
...		2	**SnapMarkerColor**	Specifies snap marker color
...		6	**SnapMarkerSize**	Specifies snap marker size
...		2	**SnapMarkerThickness**	Specifies snap marker thickness
SnapMode	0	0	SnapMode	
SnapStyl	0	0	SnapStyl	
SnapType	0	0	SnapType	
SnapUnit	0.5000,0.5000	0.5;0.5	SnapUnit	
SolidCheck	1	1	SolidCheck	
SolidHist	1		...	
SortEnts	127	96	SortEnts	
SortOrder	1		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
SplDegree	3		...	
...		0	**spaAdjustMode**	Smooths triangles
...		0	**spaGridAspectRatio**	Specifies aspect ratio of cell grids
...		0	**spaGridMode**	Specifies location of grids
...		0	**spaMaxFacetEdgeLength**	Specifies max length of a side of cell
...		512	**spaMaxNumGridLines**	Specifies max no. of grid lines in subdivisions
...		0	**spaMinUGridLines**	Specifies max no. of grid lines in u direction
...		0	**spaMinVGridLines**	Specifies max no. of grid lines in v direction
...		15	**spaNormalTol**	Specifies the normal tolerance
...		-1	**spaSurfaceTol**	Specifies maximum surface tolerance
...		1	**spaTriangMode**	Specifies which mesh is triangulated
...		1	**spaUseFacetRes**	Toggles use of FacetRes sysvar
SplFrame	0	0	SplFrame	
SplineSegs	8	8	SplineSegs	
SplineType	6	6	SplineType	
SplKnots	0		...	
SplMethod	0		...	
SplPeriodic	1		...	
...		c:\users\...	SrchPath	Specifies search paths for support files
SsFound	""	""	SsFound	
SsLocate	1	1	SsLocate	
SsmAutoOpen	1		...	
SsmPollTime	60	15	SsmPollTime	
SsmSheetStatus	2	2	SsmSheetStatus	
...		0	SsmState	Reports if Sheetset palette is open
...		0.2	**StampFontSize**	Height of plot stamp font
...		Arial	**StampFontStyle**	Name of plot stamp font
...		""	**StampFooter**	Default footer text
...		""	**StampHeader**	Default header text
...		0	**StampUnits**	Units of font size, inches or mm
StandardsViolation	2		...	
StartInFolder	c:\users\...		...	
StartMode	1		...	
Startup	0	0	Startup	
StatusBar	1		...	
StatusBarState	On		...	
StepSize	6	6	StepSize	
StepsPerSec	2	2	StepsPerSec	
...		"mechanical.cst"	StructureTreeConfig	Name of structure configuration file
SubObjSelectionMode	0		...	
SunStatus	0		...	
SurfaceAssociativity	1		...	
SurfaceAssociativityDrag	1		...	
SurfaceAutoTrim	0		...	
SurfaceModelingMode	0		...	
SurfTab1	6	6	Surftab1	
SurfTab2	6	6	Surftab2	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
SurfType	6	6	SurfType	
SurfU	6	6	SurfU	
SurfV	6	6	SurfV	
...		0	**SvgBlendedGradients**	Toggles use of blended gradients
...		".png"	**SvgDefaultImageExtension**	Specifies default file name extension
...		0	**SvgGenericFontFamily**	Specifies name of generic font family
...		""	**SvgImageBase**	Specifies path to folder for saving SVG files
...		""	**SvgImageUrl**	Specifies URL for locating SVG files
...		1	**SvgLineweightScale**	Specifies pixel width of lineweights
...		768	**SvgOutputHeight**	Specifies height in points (72 points per inch)
...		1024	**SvgOutputWidth**	Specifies width in points
...		6	**SvgPrecision**	Specifies double-floating point precision
SyscodePage	ansi_1252	ANSI_1252	SysCodePage	
SysMon	1		...	

T Variables

...		25	**TabControlHeight**	Specifies height of document tab, in pixels
TableIndicator	1		...	
TableToolbar	2		...	
TabMode	0	0	TabMode	
...		0	**TabsFixedWidth**	Forces all tabs to have the same width
Target	0.0,0.0,0.0	0.0;0.0;0.0	Target	
Taskbar	1		...	
TbCustomize	1		...	
TbShowExtended	1		...	
TbShowShortcuts	On		...	
TdCreate	2455034.61	2455035.58	TdCreate	
TdInDwg	1.24	1.16E-008	TdInDwg	
TduCreate	2455034.9	2455035.88	TduCreate	
TdUpdate	2455034.61	2455035.58	TdUpdate	
TdUsrTimer	1.24	1.16E-008	TdUsrTimer	
TduUpdate	2455034.9	2455035.88	TduUpdate	
...		c:\users\...	**TemplatePath**	Specifies path to templates folder
TempOverrides	1		...	
TempPrefix	"c:\users\..."	""	TempPrefix	
...		0	TestFlags	
TextAlignMode	9		...	
TextAllCaps	1		...	
...		0	TextAngle	Stores last-used angle for text
TextAutoCorrectCaps	1		...	
TextEditor	0		...	
TextEval	0	0	TextEval	
TextFill	1	1	TextFill	
TextOutputFileFormat	0		...	
TextQlty	50	50	TextQlty	
TextSize	0.2	0.2	TextSize	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
TextStyle	standard	STANDARD	TextStyle	
...		C:/program...	**TextureMapPath**	Specifies path to texture map folders
Thickness	0	0	Thickness	
ThumbSave	1		...	
ThumbSize	1	1	ThumbSize	
TileMode	1	1	TileMode	
...		1	TileModeLightSynch	Synchronizes lighting in all viewports
TimeZone	-8000	-8000	TimeZone	
...		1	Tips	Toggles display of grip tooltips
...		16	**ToolbarIconSize**	Size of icons on toolbars and menus
ToolPalettePath	C:\Users\...	C:\users\...	ToolPalettePath	
ToolTipMerge	0		...	
Tooltips	1	1	Tooltips	
TooltipSize	0		...	
TooltipTransparency	0		...	
TouchMode	0		...	
...		0	TpState	Reports whether Tools palette is open
TraceWid	0.05	0.05	TraceWid	
TrackPath	0	0	TrackPath	
TransparencyDisplay	1	1	TransparencyDisplay	
TrayIcons	1		...	
TrayNotify	1		...	
TrayTimeout	0		...	
TreeDepth	3020	3020	TreeDepth	
TreeMax	10000000	10000000	TreeMax	
TrimMode	1	1	TrimMode	
TrustedDomains	*.autodesk.com		...	
TrustedPaths	;		...	
TSpaceFac	1	1	TSpaceFac	
TSpaceType	1	1	TSpaceType	
TStackAlign	1	2	TStackAlign	
TStackSize	70	70	TStackSize	
...		3	TtfAsText	Toggles TTF export fonts as text or vectors

U Variables

Ucs2dDisplaySetting	1		...	
Ucs3dParaDisplaySetting	1		...	
Ucs3dPerpDisplaySetting	1		...	
UcsAxisAng	90	90	UcsAxisAng	
UcsBase	WORLD	""	UcsBase	
UcsDetect	1	1	UcsDetect	
UcsFollow	0	0	UcsFollow	
UcsIcon	3	3	UcsIcon	
...		0	UcsIconPos	Toggles non-origin UCS icon position
UcsName	""	""	UcsName	
UcsOrg	0.0,0.0,0.0	0;0;0	UcsOrg	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
UcsOrtho	1	1	UcsOrtho	
UcsSelectMode	1		...	
UcsView	1	1	UcsView	
UcsVp	1	1	UcsVp	
UcsXDir	1.0,0.0,0.0	1;0;0	UcsXDir	
UcsYDir	0.0,1.0,0.0	0;1;0	UcsYDir	
UndoCtl	53	1	UndoCtl	
UndoMarks	0	5	UndoMarks	
UnitMode	0	0	UnitMode	
UOsnap	1		...	
UpdateThumbnail	15		...	
Userl1-5	0	0	Userl1-5	
UserR1-5	0	0	UserR1-5	
UserS1-5	""	""	UserS1-5	
...		2	**UseSheetMetal**	Determines the sheet metal license type
...		0	**UseStandardOpenFileDialog**	Displays additional folder in file dialog boxes

V Variables

...		1	**VbaMacros**	Toggles enabling of VBA macros
...		Bricsys	VendorName	Reports the vendor's name
...		16.1.04 (UNICODE)	_VerNum	Reports the version number
...		235	**VersionCustomizableFiles**	Reports version number of CUI and PGP files
ViewCtr	18.9,8.7,0.0	18.9,8.7,0.0	ViewCtr	
ViewDir	0.0,0.0,1.0	10.4;4.5;0.0	ViewDir	
ViewMode	0	16	ViewMode	
ViewSize	14.65	16	ViewSize	
ViewSketchMode	0	1	...	
ViewTwist	0	1	ViewTwist	
ViewUpdateAuto	1	1	ViewUpdateAuto	
VisRetain	1	1	VisRetain	
VpControl	1		...	
VpLayerOverrides	0		...	
VpLayerOverridesMode	1		...	
VpMaximizedState	0		...	
VpRotateAssoc	1	1	VpRotateAssoc	
VsCurvatureHigh	1.0		...	
VsCurvatureLow	-1.0		...	
VsCurvatureType	0		...	
VsDraftangleHigh	3		...	
VsDraftangleLow	-3		...	
VsZebraColor1	"Rgb:255,255,255"		...	
VsZebraColor2	"Rgb:0,0,0"		...	
VsZebraDirection	90		...	
VsZebraSize	45		...	
VsZebraType	1		...	
VsBackgrounds	1		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
VsEdgeColor	byentity		...	
VsEdgeJitter	-2		...	
VsEdgeOverhang	-6		...	
VsEdges	1		...	
VsEdgeSmooth	1		...	
VsEdgeLEx	-6		...	
VsFaceColorMode	0		...	
VsFaceHighlight	-30		...	
VsFaceOpacity	-60		...	
VsFaceStyle	0		...	
VsHaloGap	0		...	
VsHidePrecision	0		...	
VsIntersectionColor	"7 (white)"		...	
VsIntersectionEdges	0		...	
VsIntersectionLtype	1		...	
VsIsoOnTop	0		...	
VsLightingQuality	1		...	
VsMaterialMode	0		...	
VsMax	119.3,59.5,0.0	1E+20,1E+20,1E+20	VsMax	
VsMin	-81.3,-42.1,0.0	-1E+20,-1E+20,-1E+20	VsMin	
VsMonoColor	"Rgb:255,255,255"		...	
VsObscuredColor	"ByEntity"		...	
VsObscuredEdges	1		...	
VsObscuredLype	1		...	
VsOccludedColor	"ByEntity"		...	
VsOccludedEdges	1		...	
VsOccludedLtype	1		...	
VsShadows	0		...	
VsSilhEdges	0		...	
VsSilhWidth	5		...	
VtDuration	750		...	
VtEnable	3		...	
VtFps	7		...	

W Variables

...		1	**WarningMessages**	Toggles use of warning messages
WhipArc	0	0	WhipArc	
WhipThread	1	3	WhipThread	
WindowAreaColor	150	5	WindowAreaColor	
WipeoutFrame	2	2	WipeoutFrame	
WmfBkgnd	off	0	WmfBkGnd	
WmfForegnd	off	0	WmfForeGnd	
...		2	WndlMain	Reports window state, maximized or other
...		0	*WndlScrl*	Toggles scroll bars
...		1	WndlStat	Toggles status bar
...		1	*WndlTabs*	Toggles layout and model tabs

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		0	WndlText	Reports text window state
...		2162.0;202.0	WndPMain	Reports top left window position
...		40.0;40.0	WndPText	Reports top left text window
...		1160.0;760.0	WndSMain	Reports main window size
...		1120.0;720.0	WndSText	Reports text window size
WorkingFolder	c:\users\...		...	
WorkspaceLabel	0		...	
WorldUcs	1	1	WorldUcs	
WorldView	1	1	Worldview	
WriteStat	1	1	WriteStat	
WsAutosave	0		WsAutosave	
WsCurrent	2D drafting & annotation	2D Drafting	WsCurrent	

X Variables

XClipFrame	2	0	XClipFrame	
XDwgFadeCtl	70	70	XDwgFadeCtl	
XEdit	1	1	XEdit	
XFadeCtl	50	50	XFadeCtl	
XLoadCtl	2	2	XLoadCtl	
XLoadPath	"c:\users\..."	"C:\Users\..."	XLoadPath	
...		5	XNotifyTime	Minutes between checks for refs
XRefCtl	0	0	XRefCtl	
XRefNotify	2	1	XRefNotify	
XRefOverride	0	0	XRefOverride	
XRefType	0		...	

Z Variables

ZoomFactor	60	60	ZoomFactor	
ZoomWheel	0	60	ZoomWheel	

Variables

...		3	3dCompareMode	
3dConversionMode	1		...	
3dDwfPrec	2		...	
3dOsMode	11		...	
3dSelectionMode	1		...	

APPENDIX C

Command Alias Cross-reference

THIS APPENDIX COMPARES THE COMMAND ALIASES DEFINED BY AUTOCAD AND BRICSCAD. The list of 303 aliases is sorted alphabetically by command name. BricsCAD uses aliases to provide a quicker way to enter commands, as well as make some commands name-compatible with other programs, such as IntelliCAD and AutoCAD. (An [ICAD] indicates the alias is compatible with IntelliCAD.)

No new aliases were added to BricsCAD V17. Both Bricsys and Autodesk are no longer updating aliases for new commands, because both CAD programs now rely on AutoComplete to minimize the number of keystrokes needed to enter command names.

You can, nevertheless, define new aliases and modify existing ones in both CAD programs through these methods:

- AutoCAD customizes aliases through the **Command Aliases** button on the ribbon's **Express Tool** tab's **Tools** panel
- BricsCAD customizes aliases through the **Customize** command's **Aliases** tab

BricsCAD saves aliases in the *default.pgp* file in the following folder locations:

Windows in folder *C:\Users\<login>\AppData\Roaming\Bricsys\BricsCAD\V17x64\en_US\Support*

Mac in folder */Users/<login>/Library/Preferences/Bricsys/BricsCAD/V17x64/en_US/Support*

Linux in folder *home/<login>/Bricsys/BricsCAD/V17x64/en_US/support*

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
A Commands			
ActRecord	arr		...
-ActStop	-ars		...
ActStop	ars		...
ActUserInput	aru		...
-ActUserMessage	-arm		...
ActUserMessage	arm		...
AdCenter	adc, content, dc, dcenter		...
Align	al	al	Align
AllPlay	aplay		...
AnalysisCurvature	curvatureanalysis		...
AnalysisDraftAngle	draftangleanalysis		...
AnalysisZebra	zebraanalysis		...
...		ap	Aperture
...		planviewint ICAD	Apparent
AppLoad	ap		...
Arc	a	a	Arc
Area	aa	aa	Area
-Array	-ar		...
Array	ar	ar	Array
-AttDef	-att	-at	-AttDef
AttDef	att, ddattdef	at, ddattdef	AttDef
...		ad	AttDisp
-AttEdit	-ate, atte		...
AttEdit	ate, ddatte, ddattext	-ate	AttEdit
AttExt	ddattext	-ax	-AttExt
...		ax, ddattext	AttExt
AttIpEdit	ati		...
B Commands			
...		backgrounds ICAD	Background
BAction	ac		...
...		ba	Base
BClose	bc		...
BcParameter	cparam		...
BEdit	be		...
...		bm	Blipmode
-Block	-b	-b	-Block
Block	b, acadblockdialog, bmake, bmod	b	Block
-Boundary	-bo	-bo	-Boundary
Boundary	bo, bpoly	bo, bpoly	Boundary
BParameter	param		...
Break	br	br	Break
BSave	bs		...
BvState	bvs		...

AutoCAD Command	AutoCAD Alias	BricsCAD Alias	BricsCAD Command
C Commands			
Camera	cam		...
Chamfer	cha	cha	Chamfer
Change	-ch	-ch	Change
CheckStandards	chk		...
Circle	c	c	Circle
-Color	-col,-colour	-col, -colour	-Color
Color	col, colour, ddcolor, ddcolour	col, colour, ddcolor, ddcolour, setcolor ICAD	Color
CommandLine	cli		...
ConstraintBar	cbar		...
ConstraintSettings	csettings		...
Copy	co, cp	co, cp	Copy
...		cl	CopyLink
...		cui	Customize
CTableStyle	ct		...
CvAdd	insertcontrolpoint		...
CvHide	pointoff		...
CvRebuild	rebuild		...
CvRemove	removecontrolpoint		...
CvShow	pointon		...
Cylinder	cyl	cyl	Cylinder
D			
DataExtraction	dx		...
DataLink	dl		...
DataLinkUpdate	dlu		...
DbConnect	dbc		...
DdEdit	ed	ed	DdEdit
DdGrips	gr	gr	DdGrips
...		se	DdSelect
DdVpoint	vp	vp, viewctl, setvpoint ICAD	DdVpoint
DelConstraint	delcon		...
Dist	di	di	Dist
Divide	div	div	Divide
Donut	do, doughnut	do, doughnut	Donut
DrawingRecovery	drm		...
DrawOrder	dr	dr	DrawOrder
DSettings	ds, ddrmodes, se	ddrmodes, rm	DSettings
DsViewer	av		...
DView	dv	dv	DView
...		dx	DxfOut
Dimension Commands			
...		dimension	Dim
DimAligned	dal, dimali	dal, dimali	DimAligned
DimAngular	dan, dimang	dan, dimang	DimAngular

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
Dimarc	dar		...
DimBaseline	dba, dimbase	dba, dimbase	DimBaseline
DimCenter	dce	dce	Dimcenter
DimConstraint	dcon		...
DimContinue	dco, dimcont	dco, dimcont	DimContinue
DimDiameter	ddi, dimdia	ddi, dimdia	DimDiameter
DimDisassociate	dda		...
DimEdit	ded, dimed	ded, dimed	DimEdit
DimJogged	jog, djo		...
DimJogline	djl		...
DimLinear	dli, dimlin, dimhorizontal,	dli, dimlin, dimhorizontal, dimrotated,	DimLinear
	dimrotated, dimvertical	dimvertical	
DimOrdinate	dor,dimord	dor, dimord	DimOrdinate
DimOverride	dov,dimover	dov, dimover	DimOverride
DimRadius	dra,dimrad	dra, dimrad	DimRadius
DimReassociate	dre		...
...		-dst	-DimStyle
DimStyle	d, dst, dimsty, ddim	d, ddim, dimsty, ds,dst, expdimstyles, setdim [ICAD]	DimStyle
DimTedit	dimted	dimted	DimTedit

E Commands

EditShot	eshot		...
...		ate	EAttEdit
Ellipse	el	el	Ellipse
Erase	e	e, delete	Erase
...		xb	ExpBlocks
Explode	x	x	Explode
-Export	-qpub		...
Export	exp	exp, dwfout	Export
ExportDwf	edwf		...
ExportDwfx	edwfx		...
ExportPdf	epdf		...
-ExportToAutocad	aectoacad		...
...		uc, dducs	ExpUcs
Extend	ex	ex	Extend
ExternalReferences	er		...
Extrude	ext	ext	Extrude

F Commands

Fillet	f	f	Fillet
Filter	fi		...
FlatShot	fshot		...

G Commands

GeographicLocation	geo, north, northdir	geo	GeographicLocation

AutoCAD Command	AutoCAD Alias	BricsCAD Alias	BricsCAD Command
GeomConstraint	gcon		...
Gradient	gd		...
...		g	Grid
-Group	-g		...
Group	g		...

H Commands

-Hatch	-h	-h, -bh	-Hatch
Hatch	h, bh	h, bh	Hatch
HatchEdit	he	he	HatchEdit
HatchToBack	hb		...
Hide	hi	hi	Hide
HidePalettes	poff		...

I Commands

...		idpoint ICAD	Id
-Image	-im		...
Image	im	im, expimages ICAD	Image
ImageAdjust	iad	iad	ImageAdjust
ImageAttach	iat	iat	ImageAttach
ImageClip	icl	icl	ImageClip
Import	imp	imp	Import
-Insert	-i	-i	-Insert
Insert	i, ddinsert, inserturl	i, ddinsert	Insert
...		insal	InsertAligned
InsertObj	io	io	InsertObj
Interfere	inf	inf	Interfere
Intersect	in	in	Intersect
IsolateObjects	isolate	isolate	IsolateObjects
...		is	Isoplane

J Command

Join	j		...

L Commands

-Layer	-la	-la	-Layer
Layer	la, ddlmodes	la, ddlmodes, explayers ICAD	Layer
LayerState	las, lman	las	LayerState
...		setlayer ICAD	LayMcur
-Layout	lo		...
Leader	lead	le, lead	Leader
Lengthen	len	len, editlen ICAD	Lengthen
...		lighting	Light
...		ll	LightList
Line	l	l, 3dline	Line

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
-Linetype	-lt, -ltype	-lt	-Linetype
Linetype	lt, ltype, ddltype	lt, ddltype, expltypes ICAD	Linetype
List	li, ls, showmat	li, ls	List
...		navvcube	LookFrom
Ltscale	lts	lts	LtScale
Lweight	lw, lineweight		...

M Commands

Markup	msm		...
MatBrowserOpen	mat, rmat	matb	MatBrowserOpen
MatchProp	ma, painter	ma	MatchProp
MaterialMap	setuv	setuv	MaterialMap
Materials	mat, rmat, finish	mat, finish, rmat	Materials
Measure	me		...
MeasureGeom	mea		...
MeshCrease	crease		...
MeshRefine	refine		...
MeshSmooth	smooth		...
MeshSmoothLess	less		...
MeshSmoothMore	more		...
MeshSplit	split		...
MeshUncrease	uncrease		...
Mirror	mi	mi	Mirror
Mirror3d	3dmirror	3m, 3dmirror	Mirror3d
MLeader	mld		...
MLeaderAlign	mla		...
MLeaderCollect	mlc		...
MLeaderEdit	mle		...
MLeaderStyle	mls		...
MLine	ml	ml	MLine
Move	m	m	Move
...		msnapshot ICAD	MSlide
MSpace	ms	ms	MSpace
-MText	-t		...
MText	mt, t	mt, t	MText
MView	mv	mv	MView

N Commands

NavSMotion	motion		...
NavSMotionClose	motioncls		...
NavSWheel	wheel		...
NavVCube	cube	navvcube	LookFrom
NewShot	nshot		...
NewView	nview		...
...		ddnew	NewWiz

AutoCAD Command	AutoCAD Alias	BricsCAD Alias	BricsCAD Command
O Commands			
Offset	o	o	Offset
...		undelete, unerase	Oops
Open	openurl, dxfin	op	Open
Options	op, preferences	cfg, config, prefs, preferences	Options
...		ortho, or	Orthogonal
-Osnap	-os	-os, esnap	-OSnap
Osnap	os, ddosnap	os, ddosnap, ddesnap, setesnap ICAD	Osnap
P Commands			
-Pan	-p	-p	...
Pan	p	p	Pan
-Parameters	-par		...
Parameters	par		...
-PartialOpen	partialopen		...
PasteSpec	pa	pa	PasteSpec
PEdit	pe	pe, editpline ICAD	PEdit
PLine	pl	pl, polyline ICAD	PLine
Plot	print, dwfout		...
PlotStamp	ddplotstamp		...
Point	po	po	Point
PointCloud	pc		...
PointCloudAttach	pcattach		...
PointCloudIndex	pcindex		...
PointLight	freepoint		...
Polygon	pol	pol	Polygon
PolySolid	psolid	pso	PolySolid
Preview	pr,pre	pre, ppreview ICAD	Preview
Properties	props, ch, mo, ddchprop, ddmodify	pr, props, ch, mo, ddchprop, ddmodify	Properties
PropertiesClose	prclose	prc	PropertiesClose
PSpace	ps	ps	PSpace
PublishToWeb	ptw		...
-Purge	-pu	-pu	-Purge
Purge	pu	pu	Purge
Pyramid	pyr	pyr	Pyramid
Q Commands			
QLeader	le		...
...		n	QNew
...		qt	QText
QuickCalc	qc		...
QuickCui	qcui		...
Quit	exit	exit	Quit
QvDrawing	qvd		...
QvDrawingClose	qvdc		...

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
QvLayout	qvl		...
QvLayoutClose	qvlc		

R Commands

Rectang	rec, rectangle	rec, rect, rectangle	Rectang
Redraw	r	r	Redraw
RedrawAll	ra	ra	RedrawAll
Regen	re	re	Regen
RegenAll	rea	rea	RegenAll
Region	reg	reg	Region
...		ri	Reinit
-Rename	-ren	-ren	-Rename
Rename	ren	ren, ddrename	Rename
Render	rr	rr	Render
RenderCrop	rc		...
RenderEnvironment	fog	fog	RenderEnvironment
RenderPresets	rp, rfileopt	roptions	RenderPresets
RenderWin	rw, rendscr	rendscr	RenderWin
Revolve	rev	rev	Revolve
Ribbon	dashboard		...
RibbonClose	dashboardclose		...
Rotate	ro	ro	Rotate
...		3r, 3drotate	Rotate3d
RPref	rpr	setrender	RPref

S Commands

Save	saveurl	sa	Save
SaveAs	dxfout		...
Scale	sc	sc	Scale
Script	scr	scr	Script
Section	sec	sec	Section
...		selgrip	SelGrips
SectionPlane	splane		...
SectionPlaneJog	jogsection		...
SectionPlaneToBlock	generatesection		...
SequencePlay	splay		...
...		ucp, dducsp	SetUcs
SetVar	set	set	SetVar
...		sha	Shade
ShadeMode	sha, shade		
SheetSet	ssm	ssm	SheetSet
ShowPalettes	pon		...
...		freehand ICAD	Sketch
Slice	sl	sl	Slice
Snap	sn	sn	Snap
Solid	so	so, plane ICAD	Solid

AutoCAD Command	AutoCAD Alias	BricsCAD Alias	BricsCAD Command
Spell	sp	sp	Spell
Spline	spl	spl	Spline
SplinEdit	spe	spe	SplinEdit
Standards	sta		...
Stretch	s	s	Stretch
...		font ICAD	-Style
Style	st, ddstyle	st, ddstyle, expstyle, expstyles, expfonts ICAD	Style
Subtract	su	su	Subtract
...		sun	SunProperties
SurfBlend	blendsrf		...
SurfExtend	extendsrf		...
SurfFillet	filletsrf		...
SurfNetwork	networksrf		...
SurfOffset	offsetsrf		...
SurfPatch	patch		...
SurfSculpt	createsolid		...

T Commands

Table	tb		...
TableStyle	ts		...
Tablet	ta	ta	Tablet
...		-t	-Text
Text	dt, dtext	tx	Text
TextEdit	tedit		...
Thickness	th	th	Thickness
TileMode	ti, tm		...
...		ti	Time
Tolerance	tol	tol	Tolerance
Toolbar	to		...
ToolPalettes	tp		...
Torus	tor	tor	Torus
Trim	tr	tr	Trim

U Commands

Ucs	dducs		...
UcsMan	uc, dducs, dducsp		...
Union	uni	uni	Union
UnisolateObjects	unhide, unisolate	unhideobjects, unhide, unisolate	UnisolateObjects
-Units	-un	-un	-Units
Units	un, ddunits	un, ddunits	Units

V Commands

...		vba	VbaIde
-View	-v	-v	-View
View	v, ddview	v, ddview, expviews ICAD	View

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
ViewGo	vgo		...
ViewPlay	vplay		...
-VisualStyles	-vsm		...
VisualStyles	vs, vsm		...
...		vl	VpLayer
VPoint	-vp	-vpoint, -viewpoint, viewpoint ICAD	VPoint
VPorts	viewports	vw, vport, viewports	VPorts
...		vs, vsnapshot ICAD	VSlide
VsCurrent	vs		...

W Commands

-WBlock	-w		...
WBlock	w, acadwblockdialog	w	WBlock
...		closeall	WCloseAll
Wedge	we	we	Wedge
...		wi	WmfIn
...		wo	WmfOut

X Commands

XAttach	xa	xa	XAttach
-XBind	-xb		...
XBind	xb	-xb	XBind
XClip	xc	clip	XClip
XLine	xl	xl, infline ICAD	XLine
-XRef	-xr	-xr	-Xref
XRef	xr	xr, expxrefs ICAD	Xref

Z Command

Zoom	z	z	Zoom

3

Commands

3dAlign	3al		...
3dArray	3a	3a, array3d	3dArray
3dFace	3f,	3f, face	3dFace
...		mesh	3dMesh
3dMove	3m		...
3dOrbit	3do, orbit		...
3dPoly	3p	3p	3dPoly
3dPrint	3dp, 3dplot, rapidprototype		...
3dRotate	3r		...
3dScale	3s		...
3dWalk	3dnavigate, 3dw		...

APPENDIX D

Keystroke & Button Cross-reference

THIS APPENDIX COMPARES THE DEFAULT SHORTCUT KEYSTROKES AND BUTTONS defined by BricsCAD and AutoCAD. The definitions are sorted into the following groups:

Keyboard shortcuts used in the drawing area

- Function keys
- **Ctrl** keys
- **Shift** keys
- Other keys

Keyboard shortcuts used in the command bar and Text window

- **Ctrl** and other keys

Mouse and tablet buttons

- Mouse buttons
- Tablet buttons
- 3D walk and fly controls
- 3D mouse controls and buttons

There are no new keystroke shortcuts or button definitions in BricsCAD V17. To learn how to customize all aspects of BricsCAD, see the *Customizing BricsCAD* ebook available for purchase from the https://www.bricsys.com/en_INTL/ Web site.

Keyboard Shortcuts for the Drawing Area

Both BricsCAD and AutoCAD define new shortcuts and buttons, and modify existing ones:

- AutoCAD uses the **Cui** command's **Keyboard Shortcuts** node
- BricsCAD uses the **Customize** command's **Keyboard** tab

FUNCTION KEYS

The following keystroke shortcuts operate in the drawing area:

AutoCAD Action	AutoCAD Command(s)	Windows & Linux Shortcut	BricsCAD Mac Shortcut	BricsCAD Command(s)	BricsCAD Action
Displays the Help dialog box	Help	F1	F1	Help	Displays the Help dialog box
Selects entire objects during subentity selection	...	Shift+F1	...	...	
Toggles between text and graphics windows	TextScr,GraphScr	F2	F2	TextScr, GraphScr	Toggles between Text and Graphics windows
Selects vertex subobjects	...	Shift+F2	Shift+F2	CommandLine CommandLineHide	Toggles the command bar
	...	Ctrl+F2	Cmd+F2	Ribbon RibbonClose	Toggles the ribbon
Toggles object snap mode	-Osnap	F3	F3	OsMode	Toggles object snap mode
Selects edge subobjects	...	Shift+F3	Shift+F3	StatBar	Toggles the status bar
Toggles 3D object snap mode	3dOsnap	F4	F4	Tablet T	Toggles tablet mode
Selects face subobjects	...	Shift+F4	Shift+F4	ScrollBar	Toggles the scroll bars
Closes the current drawing	Close	Ctrl+F4 ⊞	...	WClose	Closes the current drawing
Closes all drawings and AutoCAD	Quit	Alt+F4 ⊞	...	Quit	Closes all drawings and BricsCAD
Cycles through isoplanes	Isoplane	F5	F5	Isoplane	Cycles through isoplanes
Selects solid history	...	Shift+F5	Shift+F5	...	
Toggles dynamic UCS mode	UcsDetect	F6	F6	UcsDetect	Toggles dynamic UCS mode
Switches to the next drawing	...	Ctrl+F6 ⊞	...	...	Switches to the next drawing
Toggles display of the grid	GridMode	F7	F7	Grid T	Toggles the display of the grid
Toggles orthogonal mode	OrthoMode	F8	F8	Orthogonal T	Toggles orthogonal mode
...	...	Shift+F8 ⊞	...	VbaMan	Displays VBA Manager dialog box
Runs VBA macros	VbaRun	Alt+F8 ⊞	...	VbaRun	Displays Run BricsCAD VBA Macro dialog box
Toggles snap mode	SnapMode	F9	F9	Snap T	Toggles snap mode
Toggles polar tracking	SnapType	F10	F10	SnapType	Toggles polar tracking
Toggles object snap tracking	PolarMode	F11	F11	PolarMode	Toggles object snap tracking
...	...	Shift+F11	...	AddInMan	Displays the Add-in Manager dialog box
Opens the VBA editor	VbaIde	Alt+F11 ⊞	...	VBA	Opens the Visual Basic Editor
Toggles dynamic input	DynMode	F12	F12	QuadDisplay	Toggles the Quad cursor
...	...	Ctrl+F12	...	...	Toggles subentity selection mode

⊞ The function is provided by Windows and cannot be customized by BricsCAD

CTRL/CMD KEYS

To operate Ctrl-key shortcuts in Linus and Windows, hold down the **Ctrl** key, and the press the associated character. In Mac, hold down the Cmd key instead.

AutoCAD Action	AutoCAD Command(s)	Windows & Linux Shortcuts	Mac Shortcuts	BricsCAD Command(s)	BricsCAD Action
Overrides LockUI Selects sub-objects	...	Ctrl	Cmd	*varies*	*Depends on the currently active command*
Toggles Properties palette	Properties, PropertiesOff	Ctrl+1	Cmd+1	Properties, PropertiesOff	Toggles Properties bar
Toggles DesignCenter palette	AdCenter, AdcClose	Ctrl+2	Cmd+2	Explorer	Displays Drawing Explorer
Toggles Tools palette	ToolPalettes, ToolPalettesOff	Ctrl+3		...	
Toggles Sheet Set Manager palette	SheetSet, SheetSetHide	Ctrl+4		...	
Toggles dbConnect palette	dbConnect, dbClose	Ctrl+6		...	
Toggles Markup Set Manager palette	Markup, MarkupClose	Ctrl+7		...	
Toggles QuickCalc palette	QuickCalc, QcClose	Ctrl+8		...	
Toggles Command Line palette	CommandLine, CommandLineHide	Ctrl+9	Cmd+9	CommandLine, CommandLineHide	Toggles command bar
Toggles CleanScreen mode	CleanScreenOn,	Ctrl+0		...	CleanScreenOff
Selects all non-frozen objects	(ai_SelAll) *	Ctrl+A	Cmd+A	SelGrips All	Selects all non-frozen objects
Toggles group mode	**	Ctrl+Shift+A		...	
Toggles snap mode	SnapMode	Ctrl+B	Cmd+B	Snap T	Toggles snap mode
Copies selected objects to Clipboard	CopyClip	Ctrl+C	Cmd+C	CopyClip	Copies selected objects to Clipboard
Copies objects with base point	CopyBase	Ctrl+Shift+C	Cmd+Shift+C	CopyBase	Copies selected objects with base point
Toggles dynamic UCS	UcsDetect	Ctrl+D		...	
Switches to the next isoplane	Isoplane	Ctrl+E	Cmd+E	Isoplane	Switches to next isoplane
Toggles object snap mode	OsMode	Ctrl+F	Cmd+F	Find	Displays Find and Replace dialog box
Toggles display of the grid	GridMode	Ctrl+G	Cmd+G	Grid T	Toggles display of the grid
Toggles pick style	PickStyle	Ctrl+H	Cmd+H	PickStyle	Toggles pick style
Toggles display of open palettes	HidePalettes	Ctrl+Shift+H		...	
Cycles thru coordinate display modes	Coords	Ctrl+I	Cmd+I	Coords	Cycles through coordinate display modes
Toggles constraint inference		Ctrl+Shift+I		...	
...	...	Ctrl+J	Cmd+J	;	Repeats the last command
Displays the Hyperlink dialog box	Hyperlink	Ctrl+K	Cmd+K	Hyperlink	Displays Hyperlink dialog box
Toggles orthographic mode	OrthoMode	Ctrl+L	Cmd+L	Orthogonal T	Toggles orthographic mode
Add objects to selection set	...	Ctrl+Shift+L	Cmd+Shift+L	LookFrom	Toggles look-from viewpoint gadget
	...	Ctrl+M		;	Repeats the last command
Displays Select Template dlg box	New	Ctrl+N	Cmd+N	New	Displays the New Drawing dialog box

AutoCAD Action	AutoCAD Command(s)	Windows & Linux Shortcuts	BricsCAD Mac Shortcuts	BricsCAD Command(s)	BricsCAD Action
Displays the Select File dialog box	Open	Ctrl+O	Cmd+O	Open	Displays the Open Drawing dialog box
Displays the Plot dialog box	Plot	Ctrl+P	Cmd+P	Print	Displays the Print dialog box
Toggles Quick Properties palette	QuickProperties	Ctrl+Shift+P	Cmd+Shift+P	OpmState	Toggles the Properties bar
Closes drawings and AutoCAD	Quit	Ctrl+Q	Cmd+Q	Quit	Closes drawings and BricsCAD
Cycles through viewports	^V **	Ctrl+R	...	^V	Cycles through viewports
Saves the current drawing	Qsave	Ctrl+S	Cmd+S	QSave	Saves the current drawing
Displays Save Drawing As dlg box	SaveAs	Ctrl+Shift+S	Cmd+Shift+S	SaveAs	Displays the Save Drawing As dialog box
Toggles tablet mode	Tablet	Ctrl+T	Cmd+T	Tablet T	Toggles tablet mode
Toggles polar tracking	SnapType	Ctrl+U		...	
Pastes objects from Clipboard	PasteClip	Ctrl+V	Cmd+V	PasteClip	Pastes entities from Clipboard
Pastes objects as block from Clipboard	PasteBlock	Ctrl+Shift+V	Cmd+Shift+V	PasteBlock	Pastes entities from Clipboard as a block
...	...	Ctrl+Alt+V	Cmd+Opt+V	PasteSpec	Displays the Paste Special dialog box
Toggles selection cycling	...	Ctrl+W	Cmd+W	WClose	Closes the current drawing
Cuts selected objects to Clipboard	CutCut	Ctrl+X	Cmd+X	CutClip	Cuts selected entities to Clipboard.
Redoes the last undo	Redo	Ctrl+Y	Cmd+Y	Redo	Redoes the last undo
Undoes the last command	U	Ctrl+Z	Cmd+Z	U	Undoes the last command
Displays layout tab to the left of the current one	Layout Set	Ctrl+PageUp	...	...	
Displays layout tab to the right of the current one	Layout Set	Ctrl+PgDown	...	...	
Cancels current command	Esc	Ctrl+[	Cmd+[	^C	Cancels current command
Cancels current command	Esc	Ctrl+\	Cmd+]	^C	Cancels current command

*) AutoCAD uses an AutoLISP routine for this function.

**) AutoCAD uses an undocumented command for this function.

SHIFT KEYS

Shift keys are temporary overrides in AutoCAD that operate object snaps during commands. Shift key-combinations are not supported by BricsCAD.

AutoCAD Action	AutoCAD Command	Shortcut Keystroke	BricsCAD Command	BricsCAD Action
Toggles orthogonal mode	Ortho	Shift	Orthographic	Toggles orthogonal mode
Toggles object snap mode	OsMode	Shift+A	...	
Overrides object snap: Center	-OSnap Cen	Shift+C	...	
Disables all snapping and tracking	-OSnap Non	Shift+D	...	
Overrides object snap: Endpoint	-Osnap End	Shift+E	...	
Disables all snapping and tracking	Orthomode Osmode Snapmode Autosnap	Shift+L	...	
Overrides object snap: Midpoint	-OSnap Mid	Shift+M	...	
Overrides object snap: Endpoint	-OSnap End	Shift+P	...	
Toggles object snap tracking mode	PolarMode	Shift+Q	...	
Enables object snap enforcement	OsnapOverride	Shift+S	...	
Overrides object snap: Midpoint	-OSnap Mid	Shift+V	...	
Toggles navigation wheel	NavSWheel	Shift+W	...	
Toggles polar mode	AutoSnap	Shift+X	...	
Toggles dynamic UCS mode	UcsDetect	Shift+Z	...	
Overrides object snap: Center	-OSnap Cen	Shift+,	...	
Enables object snap enforcement	OsnapOverride	Shift+;	...	
Toggles polar mode	AutoSnap	Shift+.	...	
Toggles object snap mode	-OSnap Off	Shift+'	...	
Toggles object snap tracking mode	PolarMode	Shift+]	...	
Toggles dynamic UCS mode	UcsDetect	Shift+/	...	

OTHER KEYS

These shortcut keystrokes do not work in the Mac version of BricsCAD.

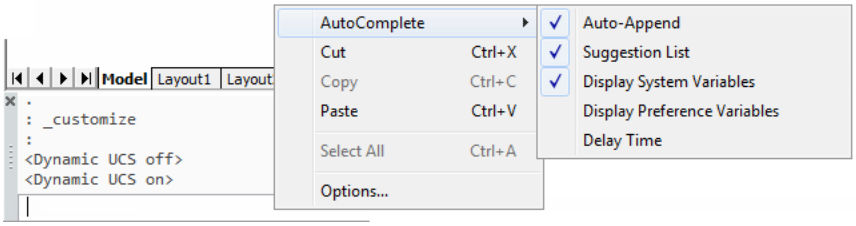
AutoCAD Action	AutoCAD Command	Shortcut Keystrokes	BricsCAD Command	BricsCAD Action
Erases selected objects	Erase	Del	Erase	Erases selected objects
	...	PageUp	Pan PgU	Pans up
	...	PageDown	Pan PgD	Pans down
	...	Shift+Left	Pan PgL	Pans left
	...	Shift+Right	Pan PgR	Pans right
	...	Shift+Up	Pan PgU	Pans up
	...	Shift+Down	Pan PgD	Pans down

Keyboard Shortcuts for Command Bar & Text Window

The following keyboard shortcuts operate on text in the command bar and Text window.

AutoCAD Action	Windows & Linux Keystroke	BricsCAD Mac Keystroke	BricsCAD Action
Executes the command or option	Enter or Spacebar	Enter or Spacebar	Executes the command or option
Repeats the previous command	Enter or Spacebar	Enter or Spacebar	Repeats the previous command
Cancels the command or option	Esc	Esc	Cancels the command or option
Displays previous command	Up	...	Displays previous command
Displays next command in command history	Down	...	Displays next command in command history
Moves cursor to the left	Left	...	Moves cursor to the left
Moves cursor to the right	Right	...	Moves cursor to the right
Moves cursor to the start of the command line	Home	...	Moves cursor to the start of the command line
Moves cursor to the end of the command line	End	...	Moves cursor to the end of the command line
Toggles between insertion and overwrite mode	Ins	...	...
Deletes characters to the right of the cursor	Del	...	...
Deletes characters to the left of the cursor	Backspace	Backspace	Deletes characters to the left of the cursor
Selects all text in Text window	Ctrl+A	Cmd+A	Selects all text in Text window
Copies selected text to Clipboard	Ctrl+C	Cmd+C	Copies selected text to Clipboard
Pastes text from Clipboard to command prompt	Ctrl+V	Cmd+V	Pastes text from Clipboard to command prompt
Cuts text from command prompt to Clipboard	Ctrl+X	Cmd+X	Cuts text from command prompt to Clipboard

As an alternative to these keystrokes, you can select text, right-click, and then choose an action from the shortcut menu.



Mouse and Tablet Buttons

The following tables compare the actions of mouse and tablet buttons in AutoCAD and BricsCAD. For BricsCAD, these buttons work identically in the Windows, Mac, and Linux versions.

MOUSE BUTTONS

AutoCAD customizes the definitions of mouse buttons in the **Mouse Buttons** and **Double-click Actions** nodes of its **CUI** command (Customize User Interface dialog box).

BricsCAD customizes mouse and double-click buttons in the **Mouse** tab of the **Customize** command (Customize dialog box).

AutoCAD Action	Mouse Button Number	BricsCAD Action
Picks objects *	1 (left button)	Picks objects *
Displays grips shortcut menu	2 (right)	Repeats the last command
Displays object snap shortcut menu	3 (center)	Displays object snap shortcut menu
Cancels the current command	4	...
Toggles snap mode	5	...
Toggles ortho mode	6	...
Toggles grid display	7	...
Changes the coordinate display	8	...
Switches to the next isoplane	9	...
Toggles tablet mode	10	...
Zooms in real time *	Wheel	Zooms in real time *
Edits selected object(s)	Double-click 1 (left button)	Edits selected object(s)
Displays object snap shortcut menu	Shift+2 (right)	Displays object snap shortcut menu
Rotates viewpoint in 3D	Shift+3 (center)	...
...	Ctrl+1 (left)	...
Displays object snap shortcut menu	Ctrl+2 (right)	Rotates viewpoint in 3D
Swivels viewpoint in 3D	Ctrl+3 (middle)	...
Zooms viewpoint in 3D	Ctrl+4	...

*) The action of the pick button (#1) and wheel cannot be customized.

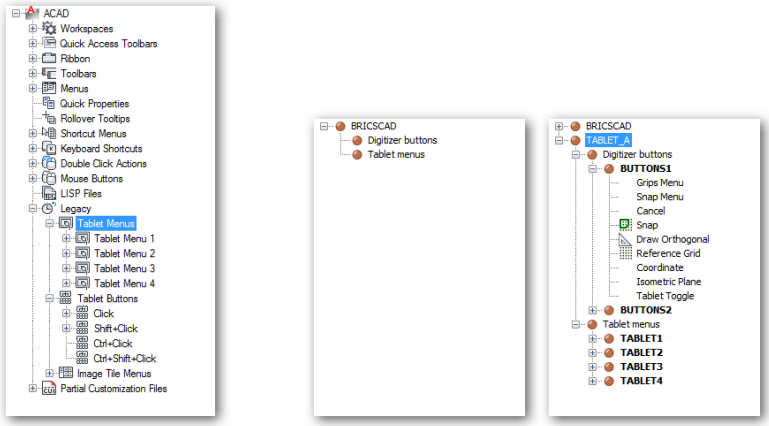
TABLET BUTTONS

AutoCAD lets you customize the definitions of stylus and puck buttons in the **Tablet Buttons** node of its Customize User Interface dialog box's **Legacy** section.

BricsCAD lets you customize buttons in the **Digitizer Buttons** node of the Customize dialog box's **Tablet** tab. However, no tablet menu or partial CUI file is provided by BricsCAD, and so the entries under Digitizer Buttons and Tablet Menus are empty, initially. The solution is to the following:

1. Download the set of partial CUI files and drawings for tablet buttons and overlays from https://www.bricsys.com/bricscad/tools/Tablet.zip.
2. Load the *tablet.cui* or *tablet(acadLike)cui* partial CUI files into BricsCAD with the **MenuLoad** command.

Notice that the two sections now contain entries for tablet buttons and menus. These work identically for the Windows, Mac, and Linux versions of BricsCAD.

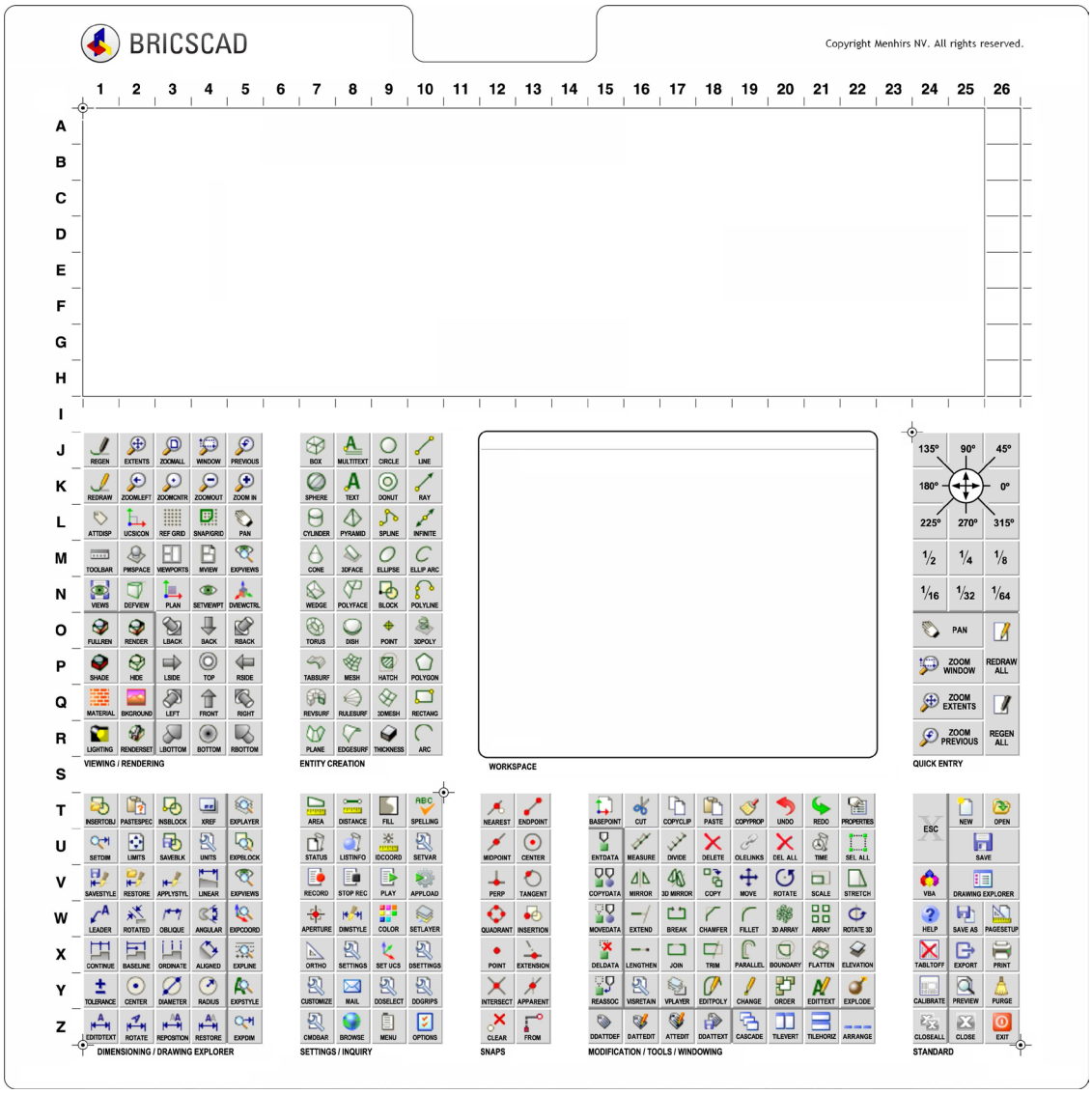


Left: *Tablet button definitions in AutoCAD's CUI dialog box.*
Center: *Default tablet definition in BricsCAD's Customize dialog box.*
Right: *Tablet definition in BricsCAD after loading "tablet(acadLike).cui."*

The following table lists the meaning of stylus and puck buttons used with tablets. Italicized text indicates the actions after partial CUI file *tablet(acadLike).cui* is loaded into BricsCAD.

AutoCAD Action	Tablet Button	BricsCAD Command	BricsCAD Action
Picks objects	1	...	Picks objects
Displays grips shortcut menu	2	$po=GRIPS $po=*	Displays grips shortcut menu
Displays object snap shortcut menu	3	$po=SNAP $po=*	Displays object snap shortcut menu
Cancels the current command	4	^c	Cancels the current command
Toggles snap mode	5	'_snap;_t	Toggles snap mode
Toggles ortho mode	6	'_orthogonal;_t	Toggles ortho mode
Toggles grid display	7	'_grid;_t	Toggles grid display
Changes the coordinate display	8	'_.COORDS $M=$(if,$(and,$(getvar, COORDS),2),0,$(+,$(getvar,COORDS),1))	Changes the coordinate display
Switches to the next isoplane	9	'_isoplane;;	Switches to the next isoplane
Toggles tablet mode	10	'_tablet;_t	Toggles tablet mode
Displays object snap shortcut menu	Shift+2	$po=SNAP $po=*	Displays object snap shortcut menu

The tablet overlay provided by Bricsys is illustrated below.



3D WALK-FLY CONTROLS

AutoCAD and BricsCAD use keystrokes and mouse buttons to control movement in 3D perspective mode, known also as "walk and fly." The keys and buttons are so different between the CAD packages that they are presented separately here. You cannot customize walk and fly controls.

AutoCAD

Enter walk or fly mode with the 3dWalk and 3dFly commands.

Function	Keystroke	Alternative Keystroke
Moves forward	w	Up-arrow
Moves backward	s	Down-arrow
Moves left	a	Left-arrow
Moves right	d	Right-arrow
Toggles between walk-fly mode	f	...
Exits walk-fly mode	Esc	Enter
Displays dialog box of keystrokes	Tab	...

BricsCAD

Enter walk mode by setting the **Perspective** system variable to 1, and then entering the **RtWalk** command.

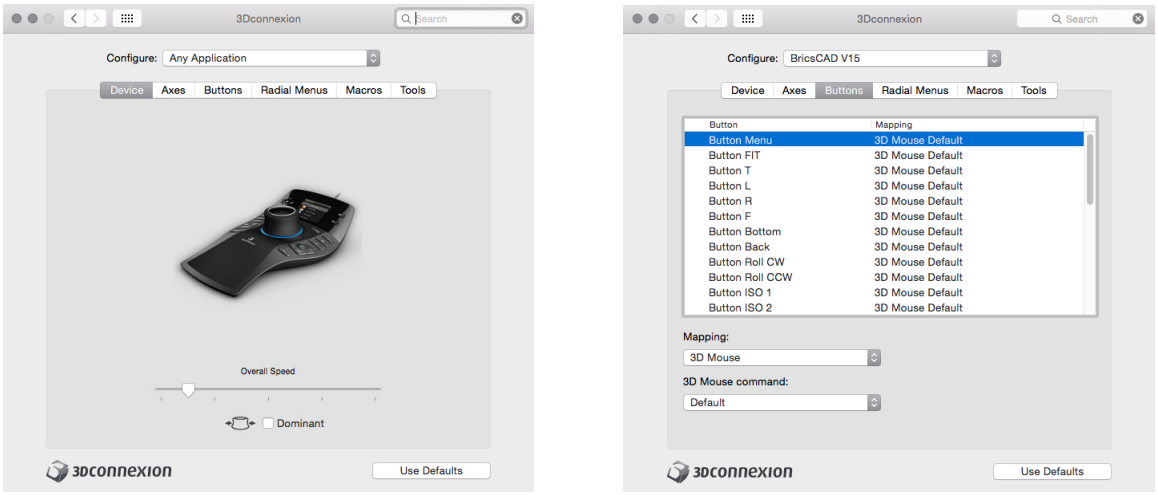
Function	Windows & Linux Button and Key	Mac Button and Key	BricsCAD Command or System Variable Executed
Moves forward, backwards, left, or right	Alt + Left button	Opt + Left button	RtWalk
Moves up, down, or sideways	Alt + Middle button	Opt + Middle button	RtUpDown
Looks around	Ctrl + Middle button	Cmd + Middle button	RtLook
Resets view direction to the horizontal	Ctrl + Home key	Cmd + Home key	...
Moves target point to the center of the scene	Alt + Home key	Opt + Home key	...
Increases walking speed	Alt + Plus key	Opt + + (plus key)	RtWalkSpeedFactor
Decreases walking speed	Alt+Minus key	Opt + - (minus)	RtWalkSpeedFactor
Increases rotation speed	Ctrl + Plus key	Cmd + + (plus)	RtRotationSpeedFactor
Decreases rotation speed	Ctrl+Minus key	Cmd + - (minus)	RtRotationSpeedFactor

3D MOUSE CONTROLS AND BUTTONS

AutoCAD and BricsCAD both support 3D mice made by 3Dconnexion. Before the CAD programs recognize the 3D mouse, the 3Dconnexion device driver must be installed on your computer. The driver software is included with the mouse, and is available for computers running recent releases of Windows, Mac, and Linux. See http://www.3dconnexion.com for support and downloads. You may need to reboot your computer after installing the 3Dconnexion driver.

BricsCAD Customization

The actions of the 3D mouse's buttons and cap are defined by the 3Dconnection Properties software. There are no controls in BricsCAD, with the sole exception of the **Ctrl3DMouse** variable, which enables and disables the 3D mouse.



Settings for multi-button SpacePilot Pro mouse

In practice, you use both mice: the regular mouse for choosing commands and picking objects, the puck of the 3D mouse for moving the viewpoint. Users typically move the regular mouse with the right hand, and the 3D mouse with the left.

The 3D mouse cannot be customized by BricsCAD's Customize | Mouse dialog box. Instead, buttons are programmed to execute BricsCAD commands through the 3Dconnection Properties software. The screen grabs illustrate the default settings of the buttons.