UPDATED FOR V16

BRICSCAD®

FOR AUTOCAD® USERS

Ralph Grabowski

BRICSYS

Payment Information

This book is covered by copyright. As the owner of the copyright, upFront.eZine Publishing, Ltd. gives you permission to make one print copy. You may not make any electronic copies, and you may not claim authorship or ownership of the text or figures herein.

By Email
Acrobat PDF format: $**31.40**
Allow for a 15MB download.

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

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

Check or Money Order
We can accept checks from the following regions of the world:
- US funds drawn on a bank with address in the USA.
- Canadian funds drawn on a bank with a Canadian address (includes GST).
- British funds drawn on a bank in Great Britain.
- Euro funds drawn on a bank located in the EU.

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 www.worldcadaccess.com/ebooks. At this Web page, editions of this book are available for BricsCAD V8 through V15. 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

This is the 9th edition and is based on BricsCAD V16
17 December 2015

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	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 on software expenditures. You'll read about the advantages of switching to BricsCAD, how it is similar to AutoCAD, and the transition issues on which to keep an eye.

The chapters here provide you detailed information on issues like differences and similarities in user interfaces, DWG-file compatibility, and even how to operate a dual-CAD design office. Useful appendices at the end of the book exhaustively cross-reference command and system variable names between the two CAD systems, as well as aliases, shortcut keystrokes, and mouse button actions.

BricsCAD V16 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 ninth edition, the book is updated to include functions added to BricsCAD V16 and AutoCAD 2016.

Welcome!

The Bricsys Benefit

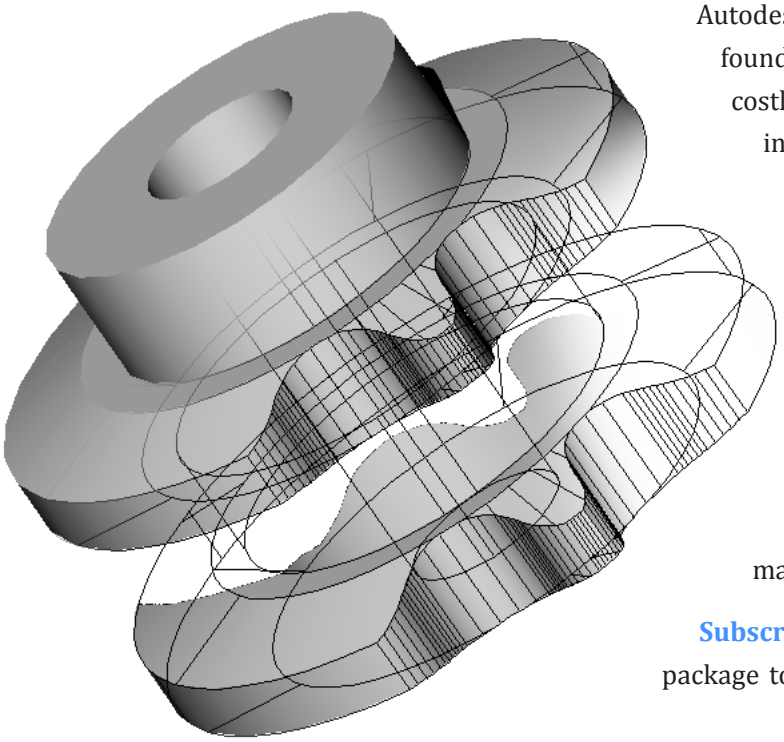
Bricsys is a small company compared to Autodesk, where executives have grown the company into a two-billion-dollar-a-year enterprise. But dealing with firms of enormous size carries a certain amount of risk.

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 as bundled with suites, such as Building Design Suite. This much choice can become confusing for potential customers determining which product or bundle is best for their need.

Given such a large number of CAD programs dependent on a single software company leads to risk when software crucial to the operation of your company might become a drag on the profits of the software company. 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; and 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), and Impressions (post-design rendering software) — among others.



Autodesk's prices tend to be high. The $4,200 price of its foundation drafting package, AutoCAD, is 4x to 10x more costly than many other 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 current CEO increases income from customers through bundles that command higher price tags (as much as $10,000) than standalone programs — and subsequently higher subscription fees — in exchange for a lower cost of each component making up the bundle.

Subscriptions. Once a customer decides which Autodesk package to license, sales people may apply pressure to also

purchase an annual support subscription. This raises TCO (total cost of ownership), and 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.

Subscription payments for AutoCAD continue to be optional at time of writing this book, but Autodesk announced that annual subscriptions will become mandatory for AutoCAD as of January 31, 2016, and that it will sell no more perpetual licences. This means that 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 AutoCAD will stop working.

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 made difficult to read through the use of uppercase letters.

If you did 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 have to erase AutoCAD. While this is meant to protect regional sales, it is shortsighted because Autodesk is blocking its customers from taking part in the reality of today's globalized business.

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-on:

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 BIM
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; Sheet Metal requires a Platinum license.

Here is a comparison of the capabilities of each edition. Functions in blue were added to BricsCAD since the last edition of this book. For a more detailed comparison, please refer to http://bricsys.com/en_INTL/bricscad/editions.

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 Modeling	Viewing	Modeling, editing, viewing	Modeling, editing, viewing
History-based 3D Modeling	Viewing	Modeling, editing, viewing	Modeling, editing, viewing
Surface 3D Modeling	Viewing	Viewing	Modeling, editing, viewing
Deformable Modeling	Viewing	Viewing	Modeling, editing, viewing
3D Assemblies and BOMs	Viewing	Viewing	Modeling, editing, viewing
Kinematic analysis	Viewing	Viewing	Modeling, editing, viewing
BOM	Viewing	Viewing	Modeling, editing, viewing
Sheet Metal Design	Viewing	Viewing	Viewing
GIS	Included	Included	Included
BIM Design	Viewing	Viewing	Modeling, editing, viewing
Rendering	...	Included	Included
Customization	Included	Included	Included
Programming	LISP, TX	LISP, TX, BRX, VBA, .Net	LISP, TX, BRX, VBA, .Net

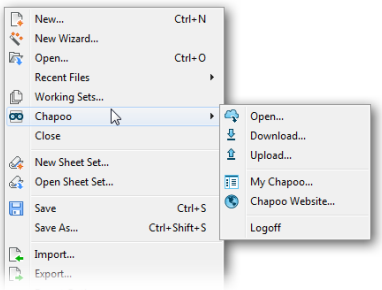
BricsCAD Platinum is the full-featured version of BricsCAD: it has everything. The Pro version is almost identical, leaving out only the parametric-based 3D modeling and 3D constraints. The Classic version costs the least because it leaves out features for which Bricsys has to pay royalties to other 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

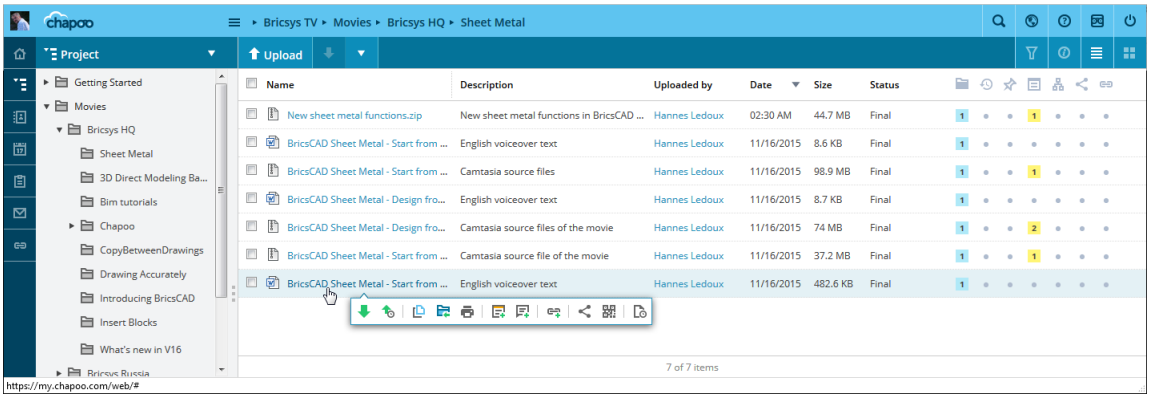
For managing drawing projects, Bricsys recommends Chapoo. This browser-based communication, collaboration, and project management system does not require BricsCAD, so it works with any office system. It is, nevertheless, integrated into BricsCAD through options in the File menu.



Chapoo 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.
- **Have 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.

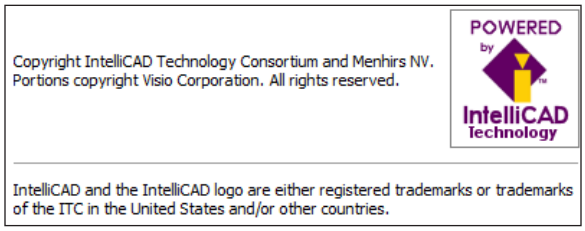


The Web interface while using Chapoo

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 www.chapoo.com.

BRICSCAD IS NOT INTELLICAD

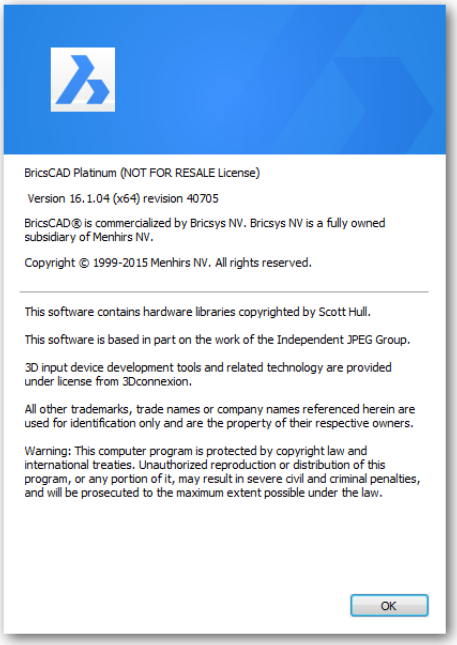
Readers familiar with BricsCAD may know it was based on IntelliCAD. *Was* is the important word here.



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 something named "Project Phoenix" from SoftDesk, which 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 its 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

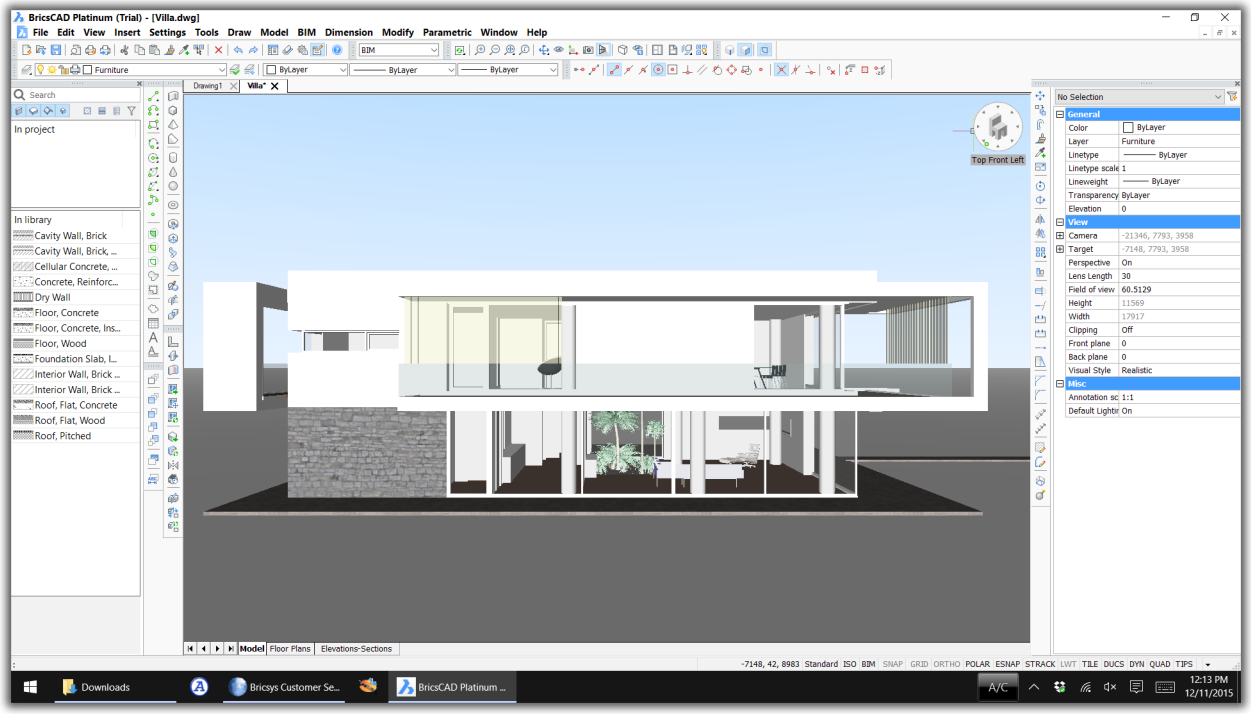
The executives of Bricsys decided they would rather develop BricsCAD on their own at a faster rate than the ITC's pace, and so made the decision to write all-new code. This huge undertaking took a couple of releases, beginning with V8. Bricsys contributed the new code to the ITC, which helped speed up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



The 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 in its Legacy workspace — complete with toolbars, menu bar, command prompt, and palettes. If you're one who doesn't care for AutoCAD's ribbon interface, then you will love BricsCAD for sticking to toolbars and menus.

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. Those that are missing are probably AutoCAD commands you weren't using, such as those for database linkages or surface modeling. 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.

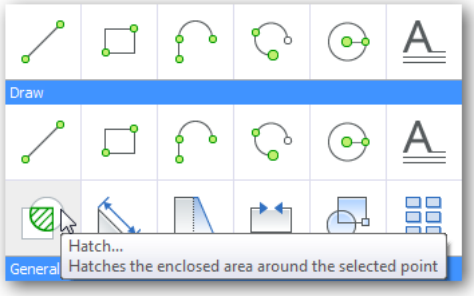
More Commands and System Variables

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

For example, all object snaps in BricsCAD have their own command names, such as Intersection and Midpoint. BricsCAD has more ways to select objects than AutoCAD, 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.

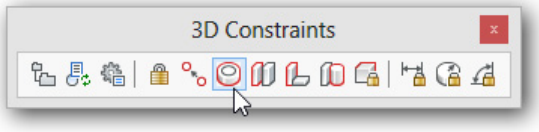
Unique to BricsCAD is the Quad cursor. When you select a feature, such as a 2D object or a 3D face, it instantly lists the most likely commands needed to manipulate the feature. As of V14, the Quad cursor can be customized. (NEW TO V16) When no objects are selected, right-click displays the Quad cursor, which the presents you with drawing commands.



The Quad cursor provides fast access to commands, many of them context-sensitive

3D Direct Modeling and Constraints

When it comes to 3D design, BricsCAD dramatically pulls ahead of AutoCAD. The Platinum edition can apply 3D constraints and infer design intent — in addition to 2D dimensional and geometric constraints. AutoCAD does not have 3D constraints at all. (The Classic and Pro editions have 2D constraints.)



BricsCAD offers 3D modeling functions not found in AutoCAD

To model assemblies of complex products, BricsCAD employs *.dwg* files of mechanical components and orders them in hierarchical structures. V16 added the ability to read 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. V14 added sheet metal design and V15 added BIM — building information modeling. V16 added None of these are in AutoCAD.

Direct modeling and editing lets you directly interact with 3D models. See chapter 6 for more. While this is possible in AutoCAD, Autodesk encourages its users to instead use the stand-alone Fusion 360 software at 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.

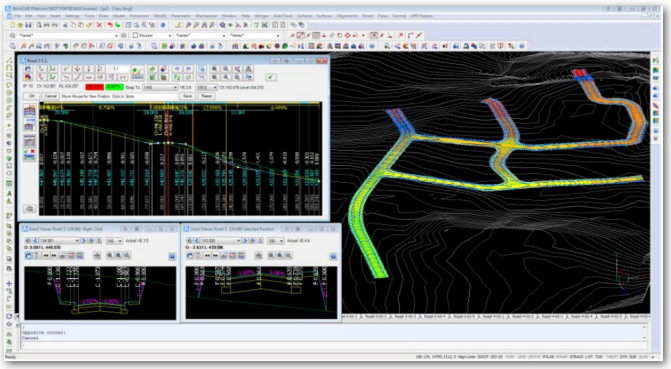
AutoCAD API	Equivalent in BricsCAD	Notes
Action Recorder (*)	Scripts, SCR	AutoCAD's Action Recorder scripts cannot be edited; scripts recorded by BricsCAD can be edited.
ActiveX	ActiveX	In-place editing; not available in BricsCAD for Linux or Mac
ADS	SDS	ADS code ported from AutoCAD requires just a recompile 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 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.

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.

Examples of Add-ons

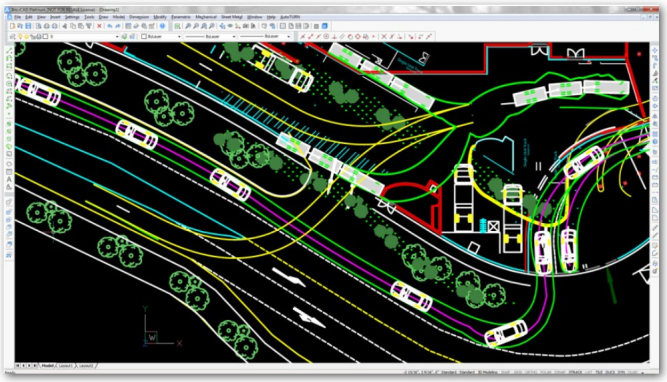
Independent programmers have written dozens of add-ons that tailor BricsCAD for specific applications in the areas of AEC, civil, data exchange, electrical, GIS, survey and mapping, general tools, HVAC, mechanical, packaging, rendering, and structural design. Here are a few examples:

Advanced Road Design from Civil Survey of Australia dynamically designs roads, drainage systems, and site preparation. www.civilsurveysolutions.com.au



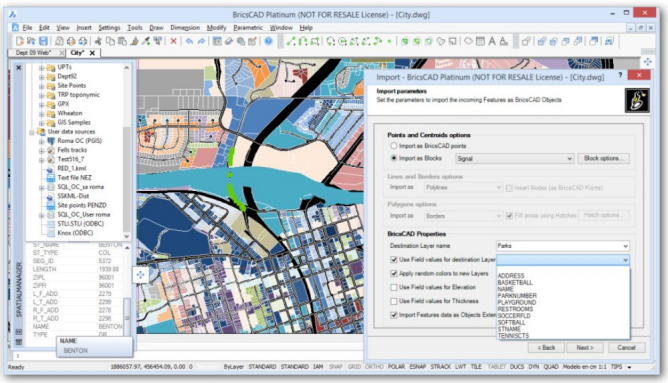
Advanced Road Design analyzing cross-sections and elevations of roads in BricsCAD

AutoTURN from Transoft Solutions of Canada is a vehicle swept-path analysis and turn simulation software for engineers, architects, and planners. www.transoftsolutions.com



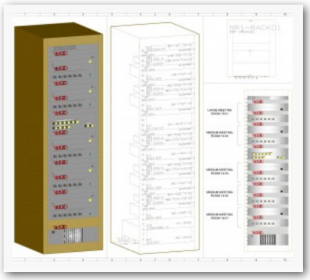
AutoTurn designing roads to accommodate the turn radii of cars and trucks in BricsCAD

Spatial Manager from Opencartis of Spain is a powerful plug-in that imports, exports, and manage spatial data quickly and simply. www.spatialmanager.com



Spatial Manager importing GIS data into BricsCAD

VidCAD Engineer from VidCAD of USA designs and documents audio, broadcast, communications, and control systems, from basic concepts and going all the way to 3D room and rack layouts. www.vidcad.com



VidCAD displaying 3D rack system in BricsCAD

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

No-charge Developer Network

Bricsys does not charge third-party developers, unlike most other CAD vendors; Autodesk charges an annual fee of $1,400 and up. You do not pay 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 decided that to become a successful CAD company by encouraging the development of many *many* add-on applications — currently 900, 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

Several years ago, Bricsys rewrote the BricsCAD code to make it independent of operating systems. The company offers versions of BricsCAD that run natively on Windows, Linux, and Mac OS X.

AutoCAD runs on Windows and OS X, but not Linux. The OS X version misses nearly half of the functions found in the Windows version (as listed by Autodesk at http://www.autodesk.com/products/autocad/compare/compare-platforms), yet the Mac version as expensive as the Windows version. BricsCAD, by contrast, boasts nearly all the same functions in all three OS versions, as shown 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 sufficient 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]	$4,195	...	$1,020	$680	$550
Upgrade [3]	...	...	$255	$255	$255
Maintenance	$ 545/year	...	$235/year	$235/year	$235/year
Subscription [4]	$1,188/year	$299/year	...	...	...

[1] US$ pricing; price may be different in other currencies.
[2] Single-user license; lower pricing available for multi-seat purchases, networked versions, and educational sites.
[3] Upgrade pricing is no longer available from Autodesk; BricsCAD's upgrade price is from the previous major release
[4] Only subscriptions are available from Autodesk during 2106; price shown is for annual renewals

Autodesk will eliminate all perpetual licenses for new purchases during 2016. This means that annual subscription payments will become compulsory when working with Autodesk. Company executives state that this change will allow them to take more revenues from their customers (see www.studiodaily.com/2014/10/autodesk-plans-to-go-subscription-only-over-next-one-to-two-years).

In contrast, BricsCAD saves you money through lower pricing, less stringent hardware requirements, and allowing you to have options, whether to upgrade (or not) or to subscribe (or not). You save even more money by switching to a free operating system, like Linux (not available from Autodesk). See Chapter 5 for running CAD on Linux.

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

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 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 www.rhino3d.com/download.htm; 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. Bricssys 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: www.bricsys.com/bricscad/tools/Bricsys-NetworkLicenseManager.msi.

Follow the advice of BricsCAD's online help when setting up the network by going to 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 www.bricsys.com/en_INTL/support/#85.

Information on how to use the Bricsys network license on a client computer: www.bricsys.com/bricscad/help/en_US/V16/BricsCAD/index.html?page=source%2FNetwork.htm. Large organizations may want to automate deployment through silent installation (Windows only): www.bricsys.com/bricscad/help/en_US/V16/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: www.reprisesoftware.com/RLM_License_Administration.pdf. License administrator and user FAQs from Reprise Software: www.reprisesoftware.com/admin/software-licensing-faq.php.

WHAT IS MISSING FROM BRICSCAD?

BricsCAD doesn't have every feature found in AutoCAD. I've put together a list of the missing things. As I update this ebook each year, I find that the list becomes shorter. Here it is as of V16:

AutoPublish	CAD standards	Database links	Dynamic blocks*
Markups	Quick properties	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 partly work, and which don't work at all.

At first glance, other AutoCAD features appear to also be missing from BricsCAD, but a second glance shows that they have near-equivalents in BricsCAD 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
eTransmit	Email drawings	eTransmit, Mail
QLeader	Leaders	DimLeader
Real-time dimensioning	Quad dimensioning	Dim...
ViewCube	LookFrom widget	LookFrom
VSTA	VBA and .Net	VBA, AppLoad

What's Missing from AutoCAD

BricsCAD Platinum V16 offers these 2D and 3D functions that are not found in AutoCAD 2014:

- 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

IT departments appreciate that BricsCAD has milder hardware and operating system requirements than does AutoCAD. This is significant to them for these reasons:

- Design firms can run BricsCAD on older computers; it is necessary neither to employ high-end hardware nor to install the special display drivers AutoCAD requires.
- BricsCAD can access more RAM and CPU speed headroom on newer computers than can AutoCAD.

RECOMMENDED HARDWARE

Autodesk and Bricsys recommend that your computer meet the following specifications. AutoCAD cannot run on any computer whose CPU lacks SSE2* and 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.

Hardware	AutoCAD	BricsCAD
CPU	2.0GHz dual-core SSE2 3GHz dual-core (Windows 7 and 8)	1GHz or faster CPU
OS X	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 OS X	2GB 3GB	256MB, plus RAM required by OS
Recommended RAM OS X	4GB or more 4GB	1GB or more
Hard Disk Space	6GB for installation	250MB for program files + 1GB free space
OS X	3GB recommended	
Monitor Resolution	1024x768 minimum 1280 x 1024 recommended	1024x768 with true color (minimum)
OS X	1280x800 minimum (OS X) 2880x1800 recommended (OS X)	1024x768 with true color minimum (OS X)
Graphics Board	32-bit 128MB (minimum) workstation-class Pixel Shader 3.0 or greater for 3D Direct3D for 3D For supported brands, see www.autodesk.com/autocad-graphicscard	Any XGA or better graphics board For rendering with supported graphics boards, see http://bricsys.com/en_INTL/support/#30a=65
OS X	Built-in graphics (OS X)	Built-in graphics (OS X)
Pointing Device	Mouse	Mouse
OS X	Apple or Microsoft mouse or trackpad (OS X)	Mouse or trackpad

*) SSE2 is short for "streaming SIMD (single instruction, multiple data) extension 2. It speeds up math functions and vector transformations used by CAD applications.

Supported Operating Systems

Bricsys supports BricsCAD running on several dialects of Linux as well as on Mac OS X and newer and older releases of the Windows operating system.

Autodesk has not announced a Linux version, and no longer supports Windows Vista. Autodesk has a version of AutoCAD for the Mac, but it is missing numerous commands and most APIs. Both CAD systems run on Windows 10,but Autodesk does not support it officially at time of writing.

Here are the operating system on which the two CAD systems run:

AutoCAD 2016	BricsCAD V16
...	Windows Vista with service pack 2
Windows 7	Windows 7
Windows 8.1	Windows 8 or 8.1
Windows 10*	Windows 10
Mac OS X v10.9 or later	Mac OSX 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.

BricsCAD operating systems: http://bricsys.com/en_INTL/support/#30a=65

AutoCAD for Windows operating systems: 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 — Linux, Mac, and Windows versions. Only Microsoft's VBA is disabled in the trial version. The size of the BricsCAD download file is 169MB, 20x smaller than AutoCAD's 3.5GB 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 V16.

WHAT'S NEW IN BRICSCAD V16

This list of new and changed BricsCAD functions was compiled from version 16.1.04. Bricsys continually updates this software. For information on functions added since this book was published, please see www.bricsys.com/common/releasenotes.jsp.

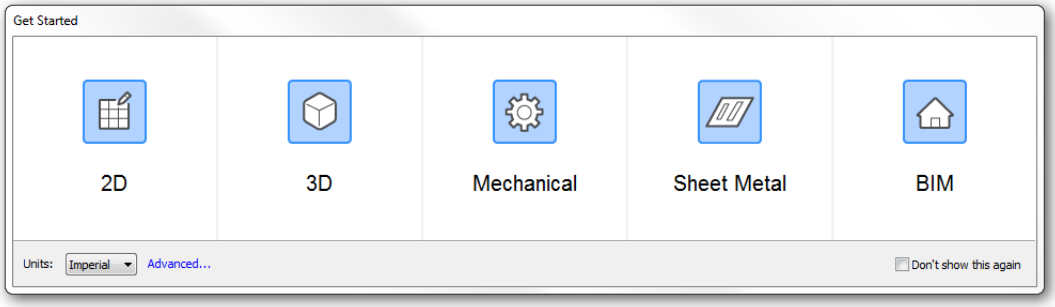
Changes are highlighted throughout this book, but be aware that the information on theses pages is not comprehensive. Command and variable names new since the last edition of this book are shown in boldface **blue** , while updated ones are shown in boldface **black**. Commands and variables are listed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

- 2D Drawing and Editing
- 3D Modeling
- BIM Design
- Communicator Module
- Files and Plotting
- Mapping
- Rendering
- Settings
- Sheet Metal Module
- Sheet Set Manager
- Text And Databases
- Libraries

USER INTERFACE

BricsCAD V16 installs and runs independently from previous BricsCAD versions.

When BricsCAD V16 starts, it displays a redesigned Getting Started dialog box; see figure below.



Redesigned getting started dialog box

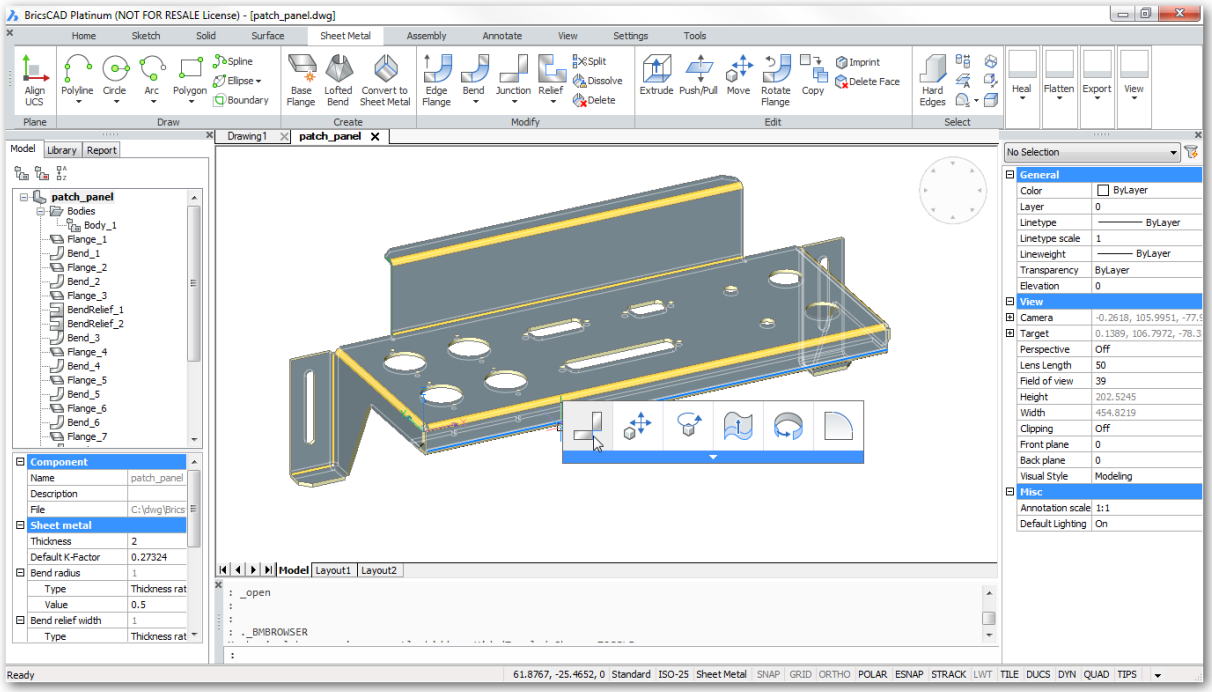
Ribbon is modified for new commands.

Mechanical Browser pane gets a new Report tab.

Sheet sets get their own pane, instead of being part of Drawing Explorer.

Other dialog boxes are updated to accommodate new functions. For example, the Layers and Print dialog boxes have options for transparency.

The Sheet Metal workspace has been added; see figure below.



New sheet metal workspace

NEW VARIABLE RELATED TO USER INTERFACE

- **DragModeInterrupt** toggles whether redraws can be interrupted; may cause incomplete graphics.

2D DRAWING AND EDITING

(dc = dimensional constraint; gc = geometric constraint)

AiMleaderEditAdd adds leader lines to multileaders.

AiMleaderEditRemove removes leader lines from multileaders.

ArrayClose saves or discards changes made by the ArrayEdit command.

-ArrayClose saves or discards changes made by the ArrayEdit command.

ArrayEdit edits the entities in associative arrays.

ArrayPath creates associative arrays by distributing entities along paths made of lines, polylines, arcs, circles, ellipses, splines, helixes, or 3D polylines.

ArrayPolar creates associative arrays by distributing entities in a circular pattern.

ArrayRect creates associative arrays by distributing entities in rows, columns, and levels.

New variables Related to Arrays

ArayAssociativity determines whether arrays are crated as associative or not.

ArayEditState reports whether an array is being edited.

dcDisplay toggles the display of dimensional constraint icons.

ESnap gets a new **GCE** geometric center mode for snapping to the centroid of closed polylines, splines, planar 3D polylines, regions, and planar faces of 3D solids.

CHANGED VARIABLE RELATED TO ESNAP

- **OsMode** flag 1024 toggles the new GCE entity snap, replaces the now-obsolete Quick esnap mode.

ExportLayout exports visible entities from the current layout to the model space of a new drawing.

-OverKill deletes duplicate entities and overlapping lines, arcs, and polylines at the command line; merges partly overlapping and contiguous ones.

Transparency property added to entities and layers, and is turned on for plotting with the Plot Transparencies option in the Page Setup and Print dialog boxes. (Plot Transparency option is turned off by default, because it takes longer to plot transparent entities.) GdiPlus supports transparency and is the default graphics interface on Windows, replacing GDI which did not support transparency.

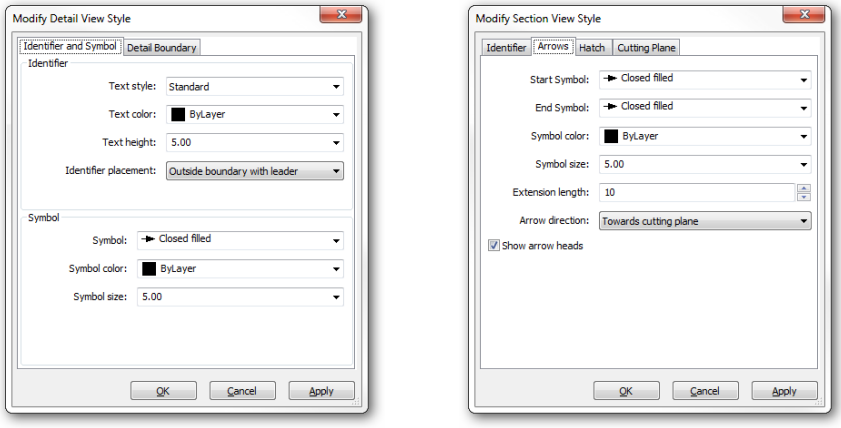
> **TIP** The **Transparency** command changes the transparency of raster images inserted in the drawing; it has no effect on other types of entities or layers, whose transparency is changed with the **Transparency** property.

NEW VARIABLES RELATED TO TRANSPARENCY

- **CeTransparency** sets the transparency for new entities.
- **HpTransparency** sets the transparency for new hatch and gradient entities.
- **TransparencyDisplay** toggles display of transparency in objects.

Trace draws wide lines, after prompting for the width; the width of each end of a trace can be changed using grips.

ViewDetailStyle specifies the style of detail views and detail symbols; see figure below.



***Left:** Dialog box for defining detail view styles; **right:** defining section view styles*

ViewEdit changes the scale and the hidden line visibility of drawing views; can be used in paper space only.

ViewExport exports drawings from paper space to model space; destroys 3D information

ViewProj generates additional projected views from existing drawing views.

ViewSectionStyle specifies the style of section views; see figure above.

3D MODELING

(bm = bricsCAD modeling; dm = direct modeling) BricsCAD V16 now supports surface modeling using existing bm- drawing and editing commands and constraints. BricsCAD V16 introduces local mechanical components, which are stored in the same .dwg file where the component is inserted in the mechanical assembly.

bmBrowser now has a Report tab in the Mechanical Browser pane for displaying the lists of sub-entities that result from executing the dmRepair and smSelectHardEdges commands.

bmExternalize converts local components to external components.

bmForm now also creates local components.

bmLocalize converts external components to local components.

bmMech now converts blocks and external reference to local and external mechanical components.

bmOpenCopy creates new drawing with a copy of selected components.

bmReplace replaces component inserts.

dmDeformCurve deforms by moving or rotating edges to a specified set of target curves.

dmDeformMove moves or rotates edges.

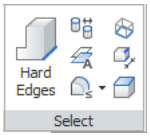
dmDeformPoint transforms points lying on specified faces.

dmGroup creates new groups, edits them, and dissolves groups.

dmRepair checks, reports, and optionally fixes errors in 3D solids.

dmSelect automatically selects similar 3D sub-entities:

- Edges of the same length
- Edges of shorter, shorter and equal, longer, or longer and equal length
- Faces of the same area
- Protrusions or depressions of the same radius
- Fillets of the same radius
- Blend networks, optionally ignoring short blends



dmSelect options in the ribbon

dmSimplify removes unnecessary edges and vertices, merges seam edges, and so on.

dmStitch converts watertight region and surface entities to 3D solids.

dmThicken converts surfaces to 3D solids with specified thicknesses.

GenerateBoundary creates closed polylines from faces of 3D solids.

Loft creates 3D solid lofts from two or more cross sections with guide curves controlling the shape, optional smooth or ruled transitions,and control of face normals and draft angles.

NEW VARIABLE RELATED TO MODELING

- **bmReportPanel** toggles whether the Mechanical Browser panel is reporting the results of commands.

BIM DESIGN

(bim = building information modeling) The BIM database now allows custom material and composition properties to be defined. To replace windows and doors, simply assign a different definition file using the File property in the properties panel.

bimAttachComposition attaches BIM compositions (wall styles) to solids.

bimDrag: now dynamically place a parallel copy when holding down the **Ctrl** key while dragging a major face of a wall and slab solids.

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 now switches between width and height of the door, and distance to neighbor-entities when holding down the **Ctrl** key. When hovering the cursor over an existing door or window, the command now places the window or door with the same parameters.

bimReposition repositions inserts (doors, windows) in the faces of solids.

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.

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

COMMUNICATOR MODULE

BricsCAD V16 is not compatible with Communicator V15, and so an upgrade to Communicator V16 is required. Communicator is an optional, extra-cost add-on that works only with Platinum editions of BricsCAD. Communiator now supports product assembly structures in models imported from ACIS, CATIA V4/V5/V6, IGES, Inventor, NX, Parasolid, Pro/E/Creo, Solid Edge, SolidWorks, STEP, VDA-FS, and XCGM). The following new variables control the import process:

NEW VARIABLES RELATED TO COMMUNICATOR

- **ImportProductStructure** determines whether structures are imported as regular blocks or as local mechanical components.
- **ImportRepair** determines whether imported geometry should be automatically repaired.
- **ImportStitch** determines whether imported geometry should be automatically stitched.
- **ImportSimplify** determines whether imported geometry should be automatically simplified.
- **ImportIgesStitch** determines whether geometry imported from IGES files should be stitched automatically.
- **ImportIgesSimplify** determines whether geometry imported from IGES files should be simplified automatically.
- **ImportSwRotateYZ** determines whether geometry imported from Solidworks files should be automatically rotated to map the Y axis to Z axis.

FILES AND PLOTTING

BricsCAD V16 now creates temporary *.dwl* and *.dwl2* drawing lock files while drawings are open to inform other users that the drawing is in use, since when, and by whom.

PdfAdjust adjusts the fade, contrast and monochrome settings of PDF underlays at the command prompt.

NEW VARIABLES

- **PdfImageAntiAlias** toggles anti-aliasing of upscaled images when saved in PDF format.
- **PdfImageCompression** specifies the compression used by images save in PDF fo4rmat.

Print now supports transparency with the Plot Transparencies option in the Page Setup and Print dialog boxes. Plot transparency option is turned off by default, because it takes longer to plot transparent entities.

-Publish publishes *.dsd* drawing sheets at the command prompt.

SupportFolder opens the *C:\Users\<login>\AppData\Roaming\Bricsys\BricsCAD\V16x64\en_US\Support* folder.

TemplateFolder opens the *C:\Users\<login>\AppData\Local\Bricsys\BricsCAD\V16x64\en_US\Templates* folder.

WhoHas display ownership information for a selected drawing file.

NEW VARIABLE

- **LocalRootFolder** specifies path to the local root folder, such as ("C:\Users\<login>\AppData\Local\Bricsys\BricsCAD\V16x64\en_US\").

MAPPING

BricsCAD V16 add projections and coordinate reference systems for New Zealand, North America, Canada, Russia and the Russian Commonwealth of Independent States.

MapConnect sets up a connection between BricsCAD and a Web-based map provider.

RENDERING

BricsCAD V16 now supports default lighting.

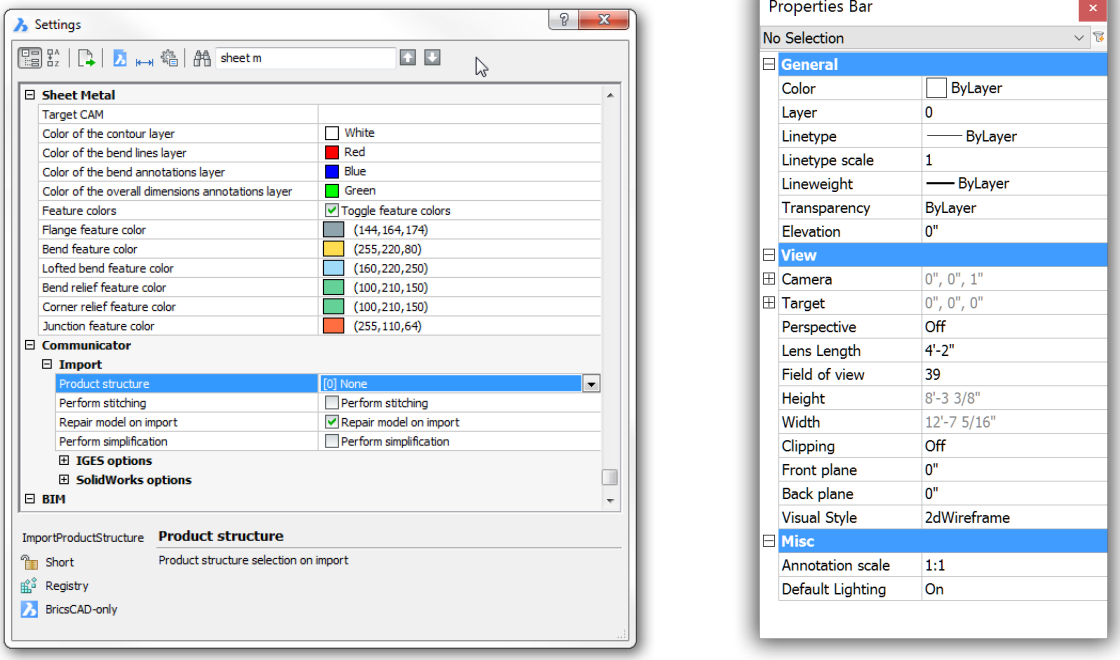
Visual styles gets a new BIM style.

NEW VARIABLES

- **DefaultLighting** determines whether default lighting is always on, or only used when no other lights are in the drawing.
- **ImageDiskCache** toggles use of the disk cache for images.
- **LinearBrightness** controls the brightness level of a viewport when default lighting is used.
- **LinearContrast** controls the contrast level of a viewport when default lighting is used.

SETTINGS AND PROPERTIES

Settings updated for new and changed system variables.



Left: Updated Settings dialog box
Right: Properties bar with additional properties

Properties palette offers the following new properties: **Transparency**, **Clipping**, **Front Plane**, **Back Plane**, and **Default Lighting**. See figure above.

SHEET METAL MODULE

(sm = sheet metal) As of BricsCAD V16, sheet metal commands are available only in a separate, extra-cost, add-on module. The sheet metal module adds the lofted bend feature with controls for the number of bends and their configuration, quadrilateral or triangular. The module supports two new corner reliefs, circular and v-type.

LicPropertiesSheetmetal reports the licence state of the sheet metal module.

SmBendSwitch converts bends to lofted bends.

smConvert now recognizes lofted bend features.

smJunctionSwitch changing symmetrical junctions to ones with overlapping faces.

smLoft constructs sheet metal bodies with lofted bends and flanges.

smReliefSwitch now changes corner relief feature between Rectangular, V-type and Circular.

smRepair joins connected lofted bends surrounded by flanges and rebuilds them tangent to adjacent flanges.

smSelectHardEdges selects all hard edges, and then reports about them in the report panel.

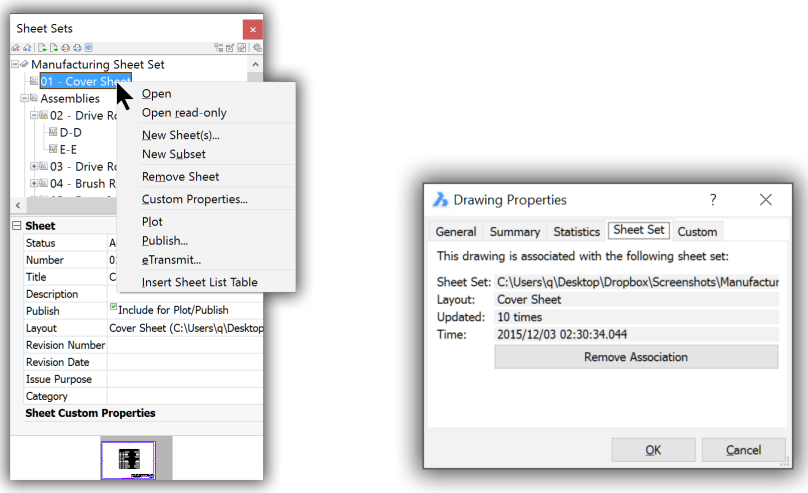
smThicken now produces lofted surfaces that can be used in sheet metal design.

NEW VARIABLES RELATED TO SHEET METAL

- **FeatureColors** toggles the assignment of colors to the faces of sheet metal features.
- **smColorBend** specifies the color of sheet metal bends.
- **smColorBendRelief** specifies the color of sheet metal reliefs.
- **smColorCornerRelief** specifies the color of sheet metal corners.
- **smColorFlange** specifies the color of sheet metal flanges.
- **smColorJunction** specifies the color of sheet metal junctions.
- **smColorLoftedBend** specifies the color of sheet metal lofted bends.
- **smLayerColorAnnotations** specifies the layer color of unfolded dimensions.
- **smLayerColorBendAnnotations** specifies the layer color of unfolded annotations.
- **smLayerColorBendLine** specifies the layer color of unfolded bend lines.
- **smLayerColorContours** specifies the layer color of unfolded 2D DXF output.
- **SmTargetCAM** specifies the intended CAM system.
- **UseSheetMetal** determines the type of sheet metal license.

SHEET SET MANAGER

(ssm = sheet set manager) BricsCAD V16 moves the Sheet Set manager from the Explorer window to its own pane.



***Left:** New Sheet Set Manager; **right:** sheet set properties in Drawing Properties dialog box*

Sheet list tables can now be placed on sheets; however, in the current implementation the table content is not associative. View label and callout blocks are placed associatively in the selected sheet view. These new options must be accessed from the Sheet Set pane, such as for the insert table option illustrated above.

DwgProps gets Sheet Set tab to report sheetset data; association can be removed with the 'Remove Association' button.

SheetsetClose closes the new Sheet Set pane, which is no longer part of the Drawing Explorer.

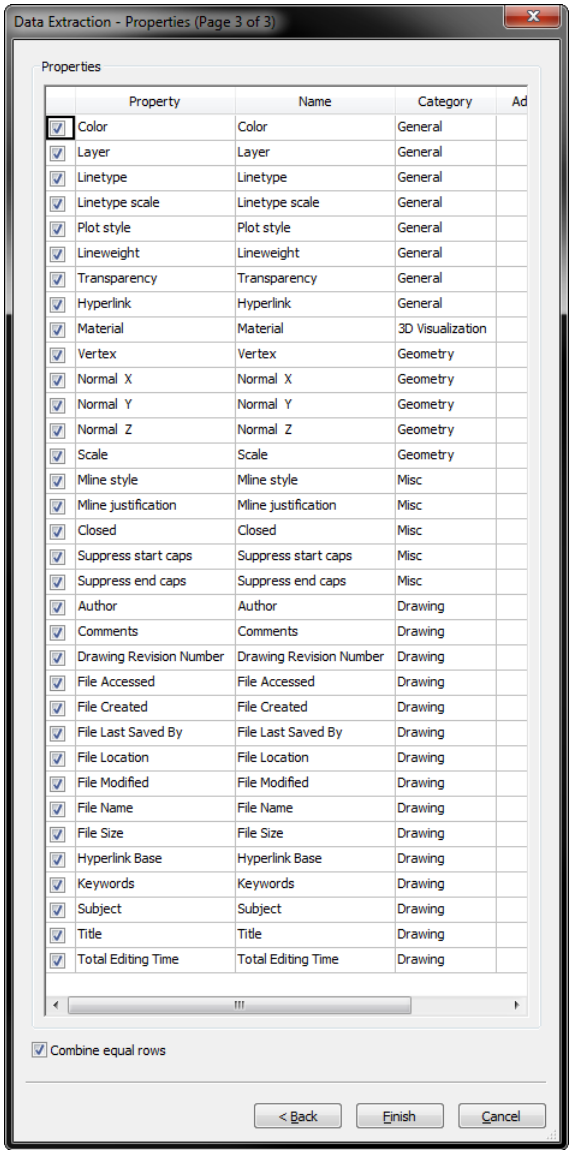
NEW VARIABLES RELATED TO SHEETSETS

- **ssmAutoOpen** determines whether the Sheet Set Manager opens automatically when drawings associated with the sheet are opened.

- **ssmPollTime** determines the refresh interval of sheet set status data.
- **ssmSheetStatus** determines how status is refreshed: never, when loaded or updated, or as specified by ssmPollTime.

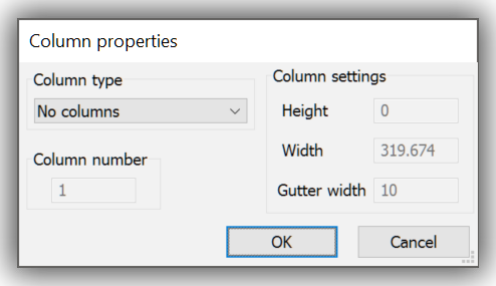
TEXT AND DATABASES

DataExtraction wizard exports almost any property of entities in the current drawing in *.csv* comma-separated values format, which can be read by Office software as well as imported into drawings as tables. The wizard's options are saved in *.dxd* data extraction definition files, which can be shared among users.



Entity properties that can be extracted from drawings.

MText now creates and edits multiple columns with options for columns that dynamically change their height or width. To access the function, click the new Column Settings button.



Options for columns in MText

LIBRARIES

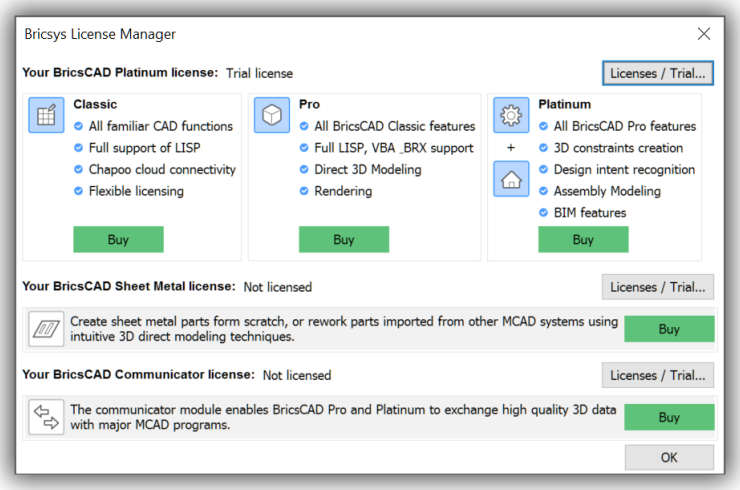
BricsCAD V16 is compiled with Visual Studio 2013 (platform toolset = v120). To be compatible, third-party C++ code with .dll extensions need to be compiled with the same V120 platform toolset.

GdiPlus supports transparency and is now the default graphics interface on Windows, replacing GDI which did not support transparency. All other visual styles — like realistic and xray — are always rendered with the RedOpenGL graphics system.

NEW VARIABLES RELATED TO GDI

- **GsDeviceType2D** selects the graphics system for wireframe displays.
- **GsDeviceType3D** selects the graphics systems for hidden, Gouraud, and Flat visual styles.

LicenseManager provides access to all Bricsys software licenses, as shown below.



License manger managing licenses

LicPropertiesSheetmetal reports the licence state of the sheet metal module.

NEW VARIABLE RELATED TO LICENCES

- **UseSheetMetal** determines the type of sheet metal license

CH-CH-CHANGES

The following commands were removed from BricsCAD since the last edition of this ebook:

Old Command	Replaced By
BmGenDraft	ViewStd command
BmGenSection	ViewSection command
HatchSetAssoc	---
Quick esnap mode	GCE (geometric center mode)
-Redefine	---
SmFlangeUnfold	SmUnfold command
smExportToOsm	smTargetCAM system variable
VsCurrent	This alias has becomes a command

Toolbars named "Mechanical" are now named "Assembly."

REMOVED VARIABLES

The following variables were removed from BricsCAD:

Removed Variable	Replaced by
GsDeviceType	GsDeviceType2D and GsDeviceType3D
Handseed	---
HomeGradientColorBottom	---
HomeGradientColorMiddle	---
HomeGradientColorTop	---
HomeGradientMode	---
R12SaveAccuracy	---
R12SaveDeviation	---
RedHilitePartial_UnselectedEdge_Alpha	---
RedHilitePartial_UnselectedEdge_Color	---
RedHilitePartial_UnselectedEdge_Smoothing	---
RedHilitePartial_UnselectedEdge_Thickness	---
SvgHiddenLineRemoving	---
UseOuputFolder	---
WorkspaceSecurity	---
XDia	---

CHAPTER TWO

Comparing User Interfaces

BRICSCAD LOOKS A LOT LIKE AUTOCAD, AS ILLUSTRATED ON THE NEXT PAGE. AS IN 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.

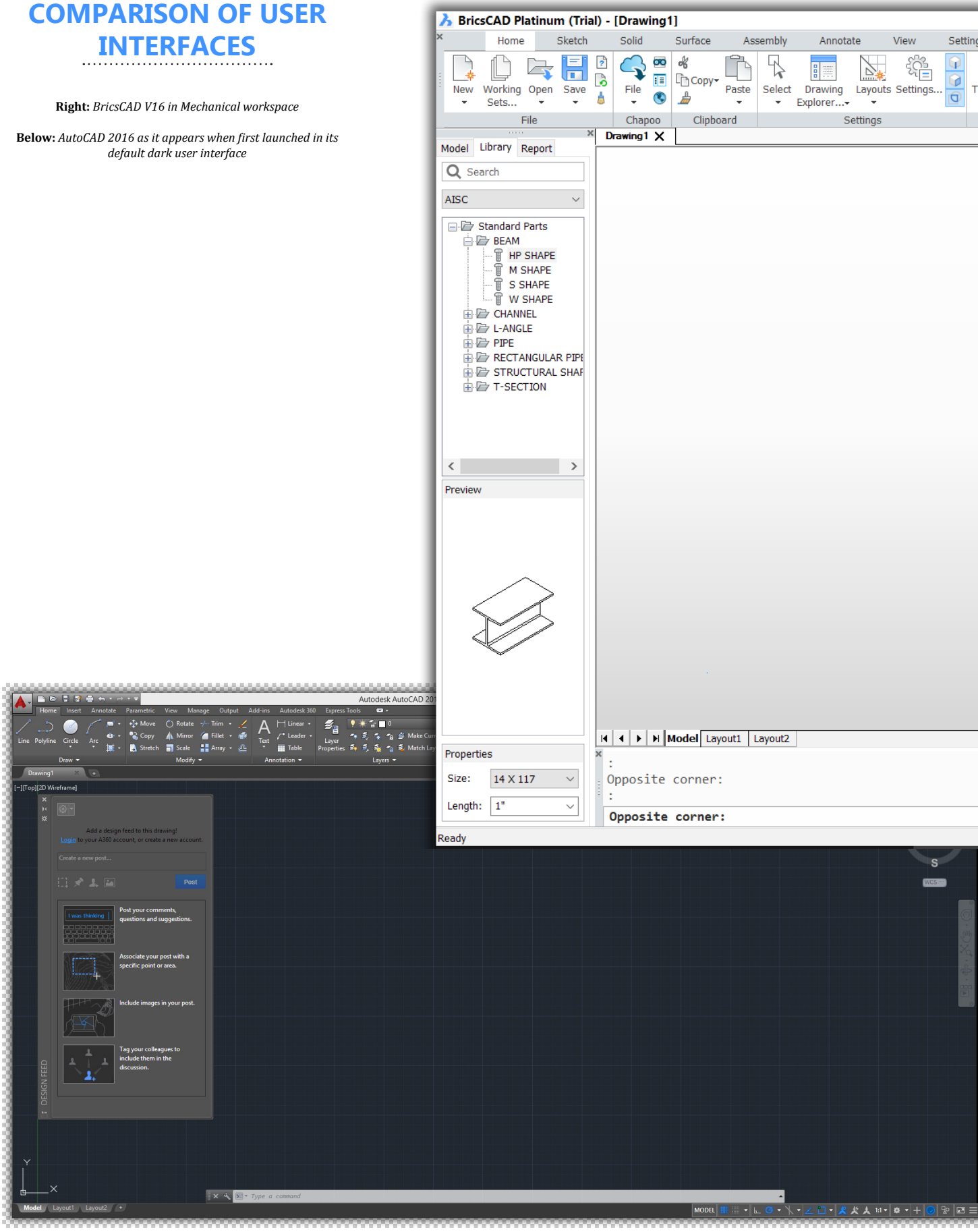
In this chapter, you learn about the similarities (and differences) between the user interfaces of the two CAD systems, specifically in the following areas:

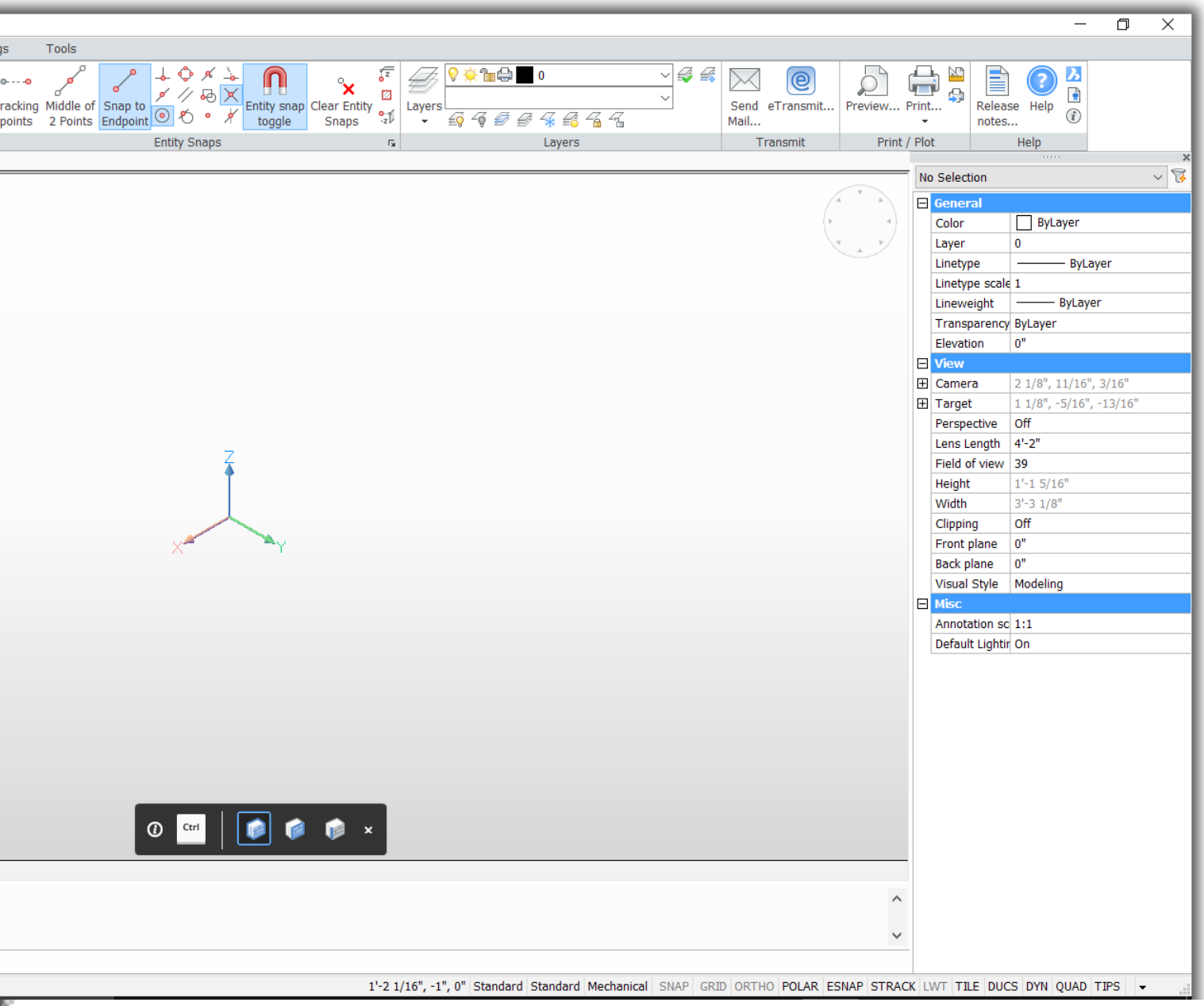
- Start screens
- Command lines and prompts
- Prompt menus (BricsCAD only)
- Quad cursor (BricsCAD only)
- Settings (BricsCAD) and Options (AutoCAD) dialog boxes
- Properties, Tool, and Sheet Set palettes
- Mechanical Browser (BricsCAD) and Parametrics Manager (AutoCAD)
- Status bar
- Selection sets
- 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 V16 in Mechanical workspace*

Below: *AutoCAD 2016 as it appears when first launched in its default dark user interface*



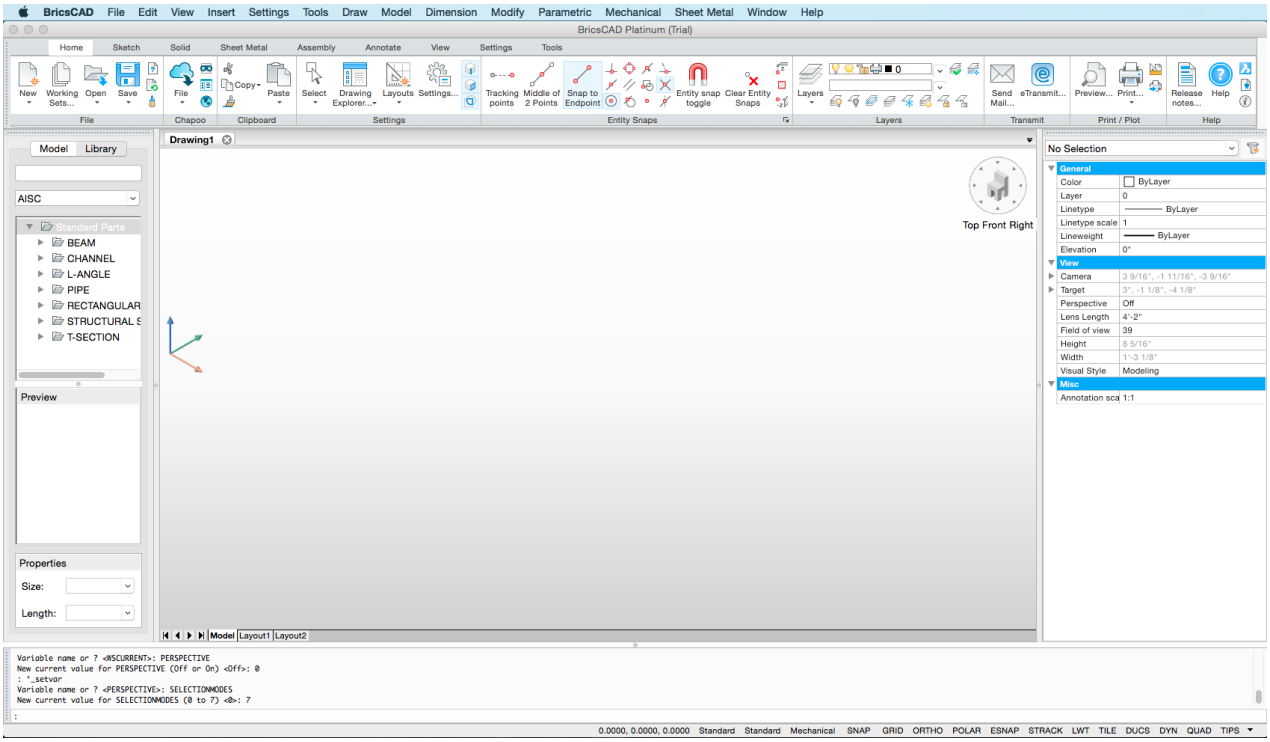

Tools
Middle of 2 Points
Snap to Endpoint
Entity snap toggle
Clear Entity Snaps
Entity Snaps
Layers
Layers
Send Mail...
eTransmit...
Transmit
Preview...
Print...
Print / Plot
Release notes...
Help
Help
No Selection
General
Color
ByLayer
Layer
0
Linetype
ByLayer
Linetype scale
1
Lineweight
ByLayer
Transparency
ByLayer
Elevation
0"
View
Camera
2 1/8", 11/16", 3/16"
Target
1 1/8", -5/16", -13/16"
Perspective
Off
Lens Length
4'-2"
Field of view
39
Height
1'-1 5/16"
Width
3'-3 1/8"
Clipping
Off
Front plane
0"
Back plane
0"
Visual Style
Modeling
Misc
Annotation sc
1:1
Default Lightir
On
Ctrl
1'-2 1/16", -1", 0"
Standard
Standard
Mechanical
SNAP
GRID
ORTHO
POLAR
ESNAP
STRACK
LWT
TILE
DUCS
DYN
QUAD
TIPS

SUMMARY OF USER INTERFACE ELEMENTS

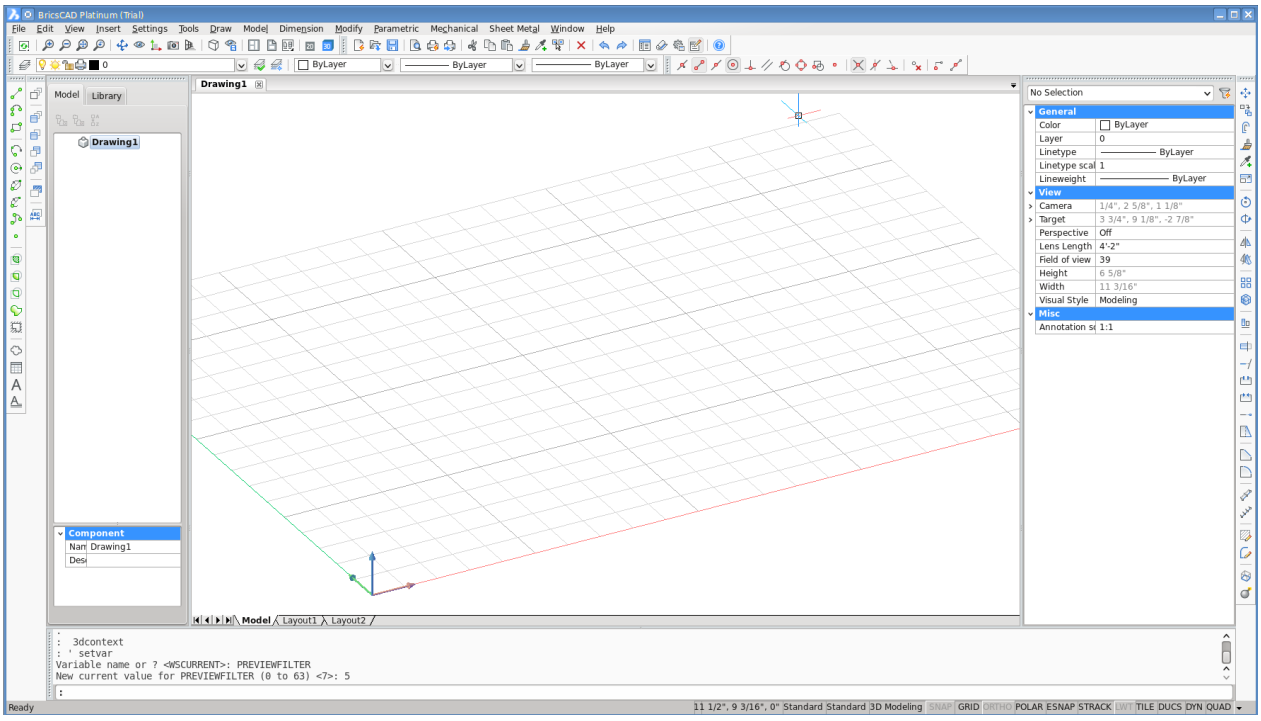
The UI elements discussed in this chapter are shown in **boldface**.

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	...
QuickView layouts and drawings	...
Drawing tabs	Drawing tabs
Ribbon	Ribbon
On the Drawing Screen	
...	**Quad cursor**
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
Steering wheels	...
Navigation cube	**Look From widget**
...	**Tips** widget for shortcut keystrokes
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	...
Quick Properties palettes	...
Quick Access toolbar	Quick Access toolbar

BricsCAD for Mac and Linux



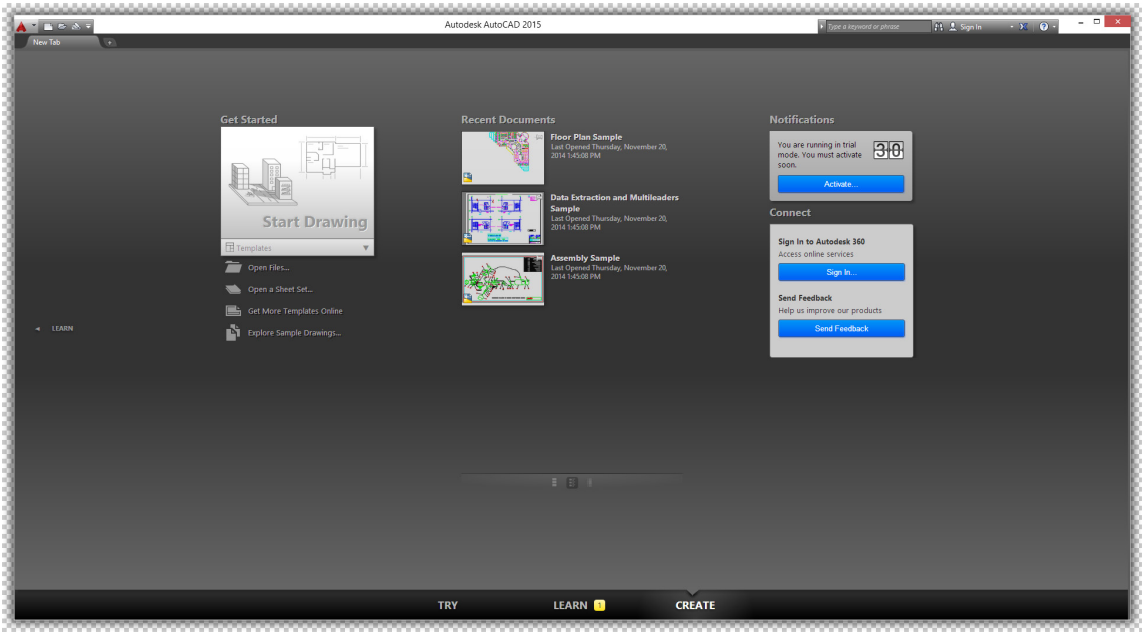
BricsCAD V15 for Mac



BricsCAD for Linux (V14 current at time of writing)

Start Screens

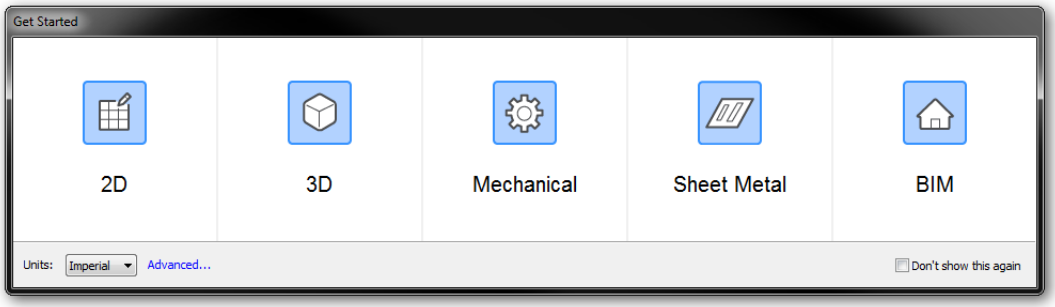
BricsCAD and AutoCAD launch with start screens. AutoCAD's is a complex, three-page affair, as illustrated below:



One of the three pages in the start screen in AutoCAD 2015

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 display black text on a light background. Screen grabs of AutoCAD in this book reflect the default setting.

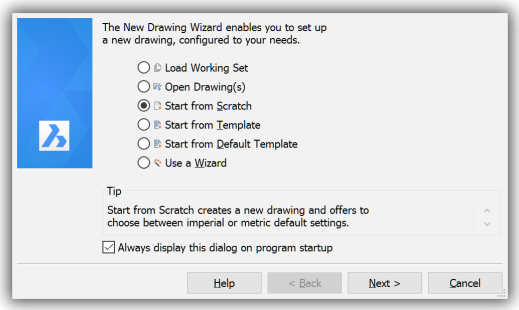
The one for BricsCAD is simple to understand. (NEW TO V16: Redesigned interface, and Sheet Metal option added.)



The new, simplified start screen in BricsCAD V16

Clicking one of the panes launches BricsCAD in the specified workspace.

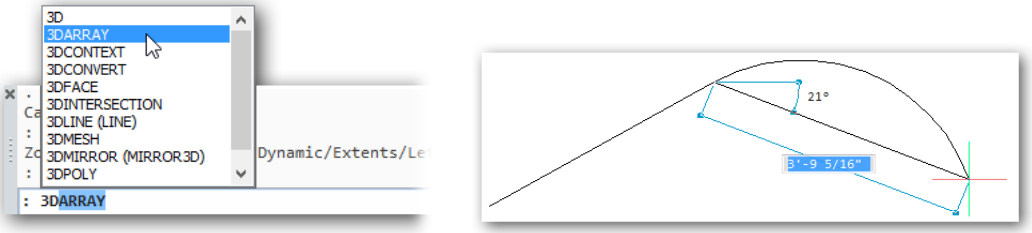
The Advanced link takes you to the Start Up screen that greeted you in earlier releases of BricsCAD: All of the options are probably familiar to AutoCAD users, except for "Load working set." It loads a group of two or more of previously opened drawings. (NEW TO V16: Start from Default Template option.)



The Startup dialog box with more specific options

Variations in User Interface

BricsCAD and AutoCAD sport user interfaces that look similar to each other, especially in AutoCAD's legacy mode. Both have toolbars, a menu bar, and a status bar. For command input, both provides users with autocomplete, dynamic input, palettes, shortcut menus, and so on. In the figure below, you see BricsCAD running with autocomplete and dynamic input.

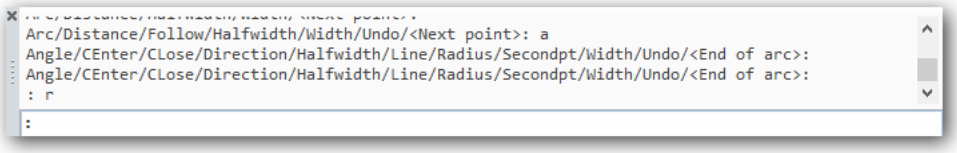


Left: *BricsCAD command bar with AutoComplete*
Right: *Dynamic input in BricsCAD drawing area*

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 them in a bit of detail.

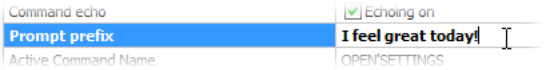
':' VS 'TYPE A COMMAND'

For its command prompt, BricsCAD uses a simple ':' to indicate it is ready for you to enter a command. Older releases of AutoCAD used 'Command:', and newer releases use 'Type a command'.



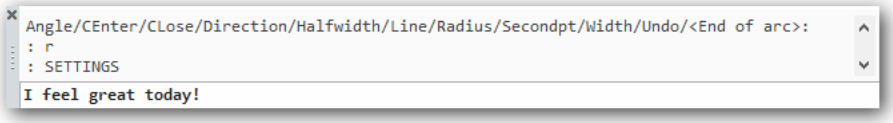
Bricsys command prompt consists of a : (colon)

If you prefer AutoCAD's prompt wording or anything else, you are free to change what BricsCAD displays. You open the Settings dialog box like this: from the **Settings** menu, choose **Settings**. 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 something silly.



Changing the prompt displayed by the command bar in BricsCAD

After you exit the dialog box, the new prompt text appears in the BricsCAD command bar right away.

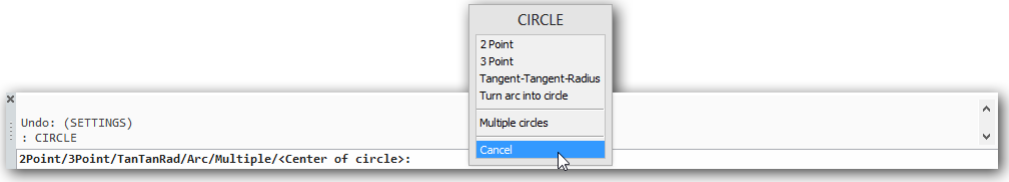


BricsCAD command prompt changed

(Command prompt customization is not available in AutoCAD.)

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: *The command bar in BricsCAD displaying options of the Circle command*
Center: *The prompt menu displaying the equivalent options*

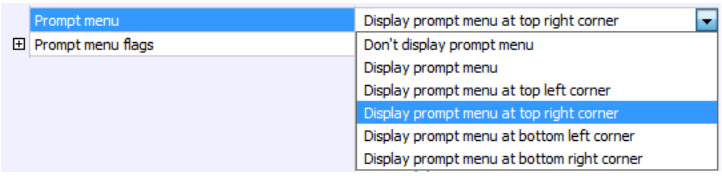
In the figure, you see command line window (at left) and the prompt menu to the right. The Circle command is in progress, and the prompts match between the command bar and the prompt menu.

You are free to use the keyboard to type in option abbreviations through the command bar, or employ your mouse to choose among options on the prompt menu. To cancel a command in progress, press **Esc** or click **Cancel**.

In certain 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

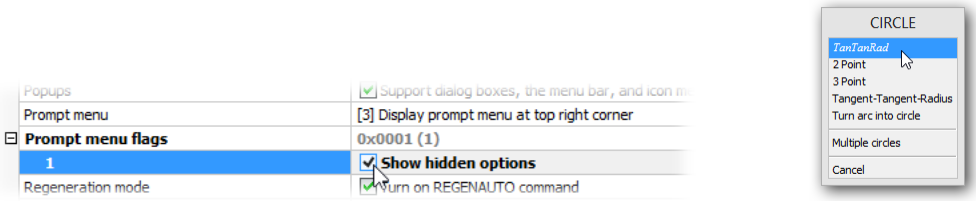
The prompt menu can be turned off. You turn the prompt menu on and off and specify its location on the screen through the Settings dialog box. In its search field, enter 'prompt menu', and then choose a setting:



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.

The **Prompt Menu Flags** option displays additional option names that normally are hidden. When displayed, these extra 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*

More Command Options

You may have noticed that the prompt menu of BricsCAD's Circle command contained more prompts than does AutoCAD's command. This is not uncommon in BricsCAD, for it has additional commands, options, and system variable useful to drafters, but 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)

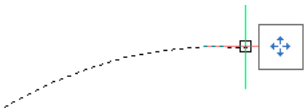
AutoCAD has workarounds to compensate for the options missing from its Circle command. To convert an arc into a circle, for instance, use its Join command; to draw multiple circles during one command, use its Multiple modifier. (Both of these are available also in BricsCAD.)

THE QUAD CURSOR (BRICSCAD ONLY)

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

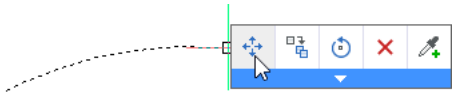
Normally, the Quad cursor is not visible; instead, most of the time you see the standard tri-color crosshair cursor. (NEW TO V16) When you right-click an empty part of the drawing, BricsCAD displays the cursor with drawing and inquiry commands. (If it does not appear, then turn on the Quad cursor by clicking **QUAD** on status bar or pressing the **F12** function key.)

The Quad cursor also does editing. When you pass the cursor over an entity, however, the Quad cursor appears initially in the form of a single button.



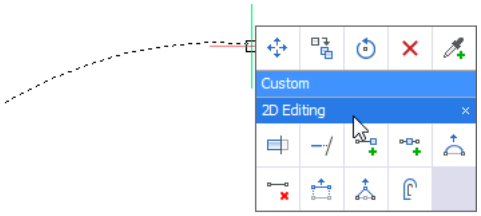
BricsCAD's Quad cursor appearing next to the crosshair cursor

At first, the Quad cursor consists of a single button that displays the icon of the last-used command. In the figure above, you see the Move command's icon. When you move the arrow cursor into the sole button, the Quad cursor expands to show additional buttons, usually for commands most commonly used with the nearest entity.



BricsCAD's Quad cursor expanding as the cursor moves over it

In addition, there are blue bands for groups of additional buttons. 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 its command.



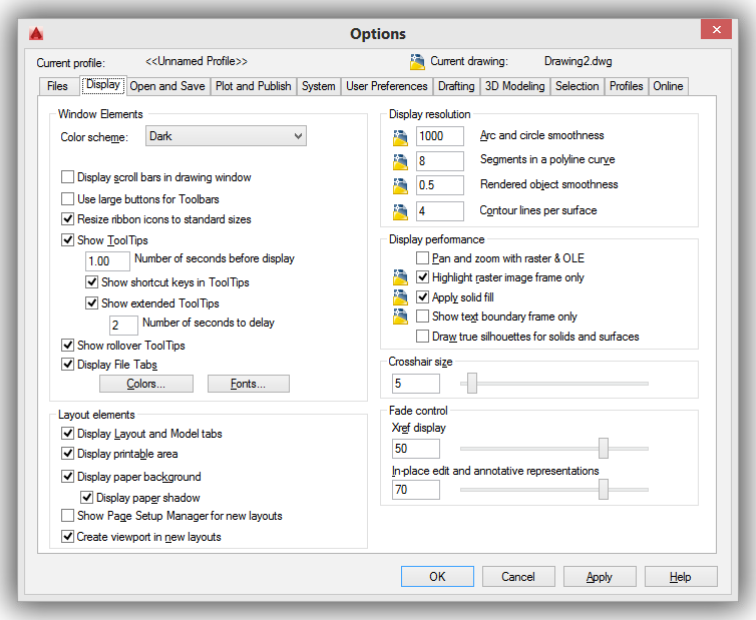
BricsCAD's Quad cursor expanding further to expose groups

BricsCAD comes with several sets of predefined Quad cursors setups, such as for 2D drafting and 3D modeling. You can customize the Custom section of the Quad cursor through the Quad and Workspace tabs in the Customize dialog box (**Tools | Customize**).

Differences in Options & Settings

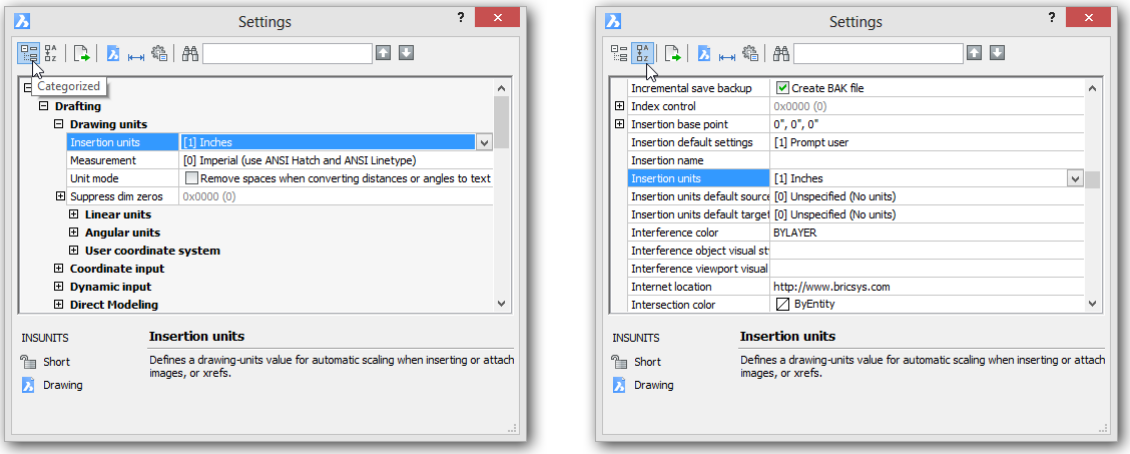
The Options dialog box in AutoCAD provides access to many system variables, but not all of them. In BricsCAD, the equivalent is known as the **Settings** dialog box and it goes to the logical conclusion: access *all* 1000+ variables. See Appendix B for the complete list in BricsCAD and a comparison with 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 eleven tabs and 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: *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.

BRIEF TOUR OF SETTINGS DIALOG BOX

Because this dialog box is really important in using BricsCAD effectively, and because this dialog box is designed differently from AutoCAD's, allow me to give you a tour of its functions. To access the Settings dialog box:

- Enter the **Settings** command
- Type the **Options** alias used by AutoCAD
- From the **Settings** menu, choose **Settings**

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

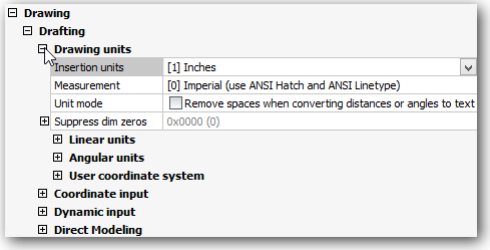
Toolbar atop the Settings dialog box

- 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

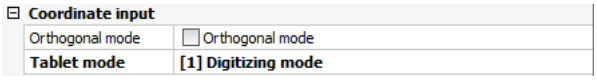
To access variables, use a method described above or else click the ⊞ boxes called "nodes" to open individual sections. (Click ⊟ boxes to close sections.) AutoCAD's CUI dialog box uses a similar system of nodes.



Opening and closing nodes to see and hide sections

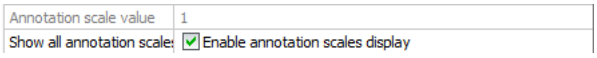
Accessing and Understanding Values

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



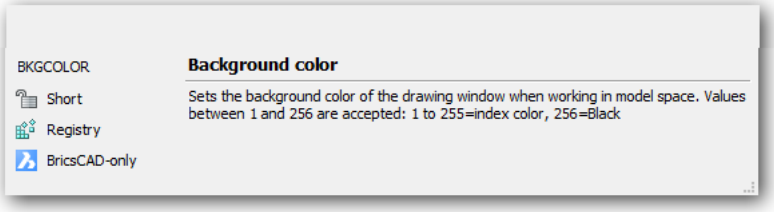
Boldfaced values have been changed since the dialog box was opened

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:

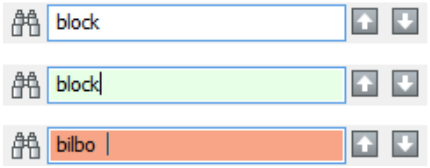


BricsCAD explaining the meaning of variables

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

Using Realtime Search

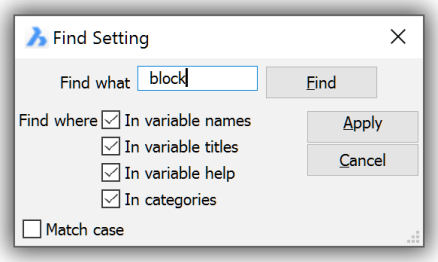
The realtime search field lets you directly access system variables by name. As you enter the first few letters, BricsCAD immediately jumps to the first name that matches them. You can then click the left and right arrows to move back and forward through matching candidates. (AutoCAD does not have a 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

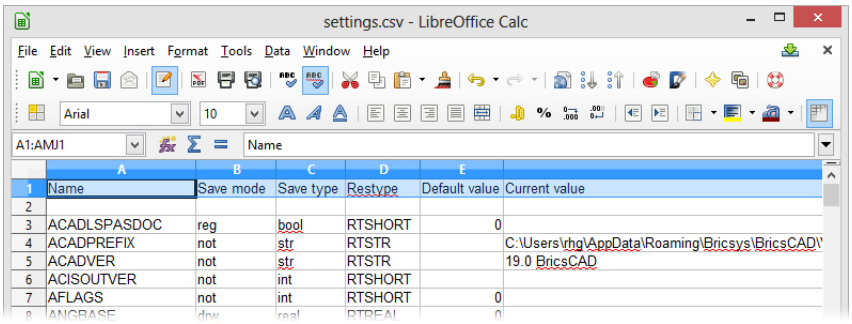
By clicking the **Find** button, you can ask BricsCAD to narrow the search, but I find it's best to leave all the Find Where options turned on.



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.)



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 "bar" in place of palette. Here is the list of bar-palettes provided:

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

In the following sections, we look at the Properties, Tools, Sheet Sets, Mechanical Browser, and Status bar.

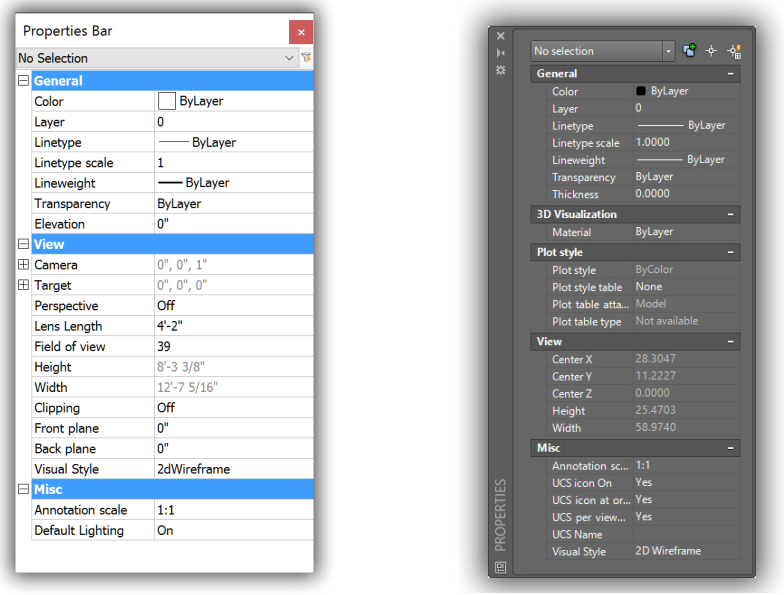
DIFFERENCES IN PROPERTIES PALETTES

The two CAD packages share a similar-looking Properties palette, except that BricsCAD calls its the Property "bar." It operates just like the Properties palette in AutoCAD, but with this important difference: BricsCAD employs the Properties bar for editing objects and properties, whereas AutoCAD would tend to display a command-specific dialog box or bring up a contextual tab on the ribbon.

For example, when you click on a hatch pattern in BricsCAD, the Properties bar displays all the options you expect to find in AutoCAD's Hatch Edit dialog box or contextual ribbon.

To turn on the Properties bar in BricsCAD, enter the **Properties** command or chose **Modify|Properties** from the menu. It appears automatically when you double-click entities in drawings.

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

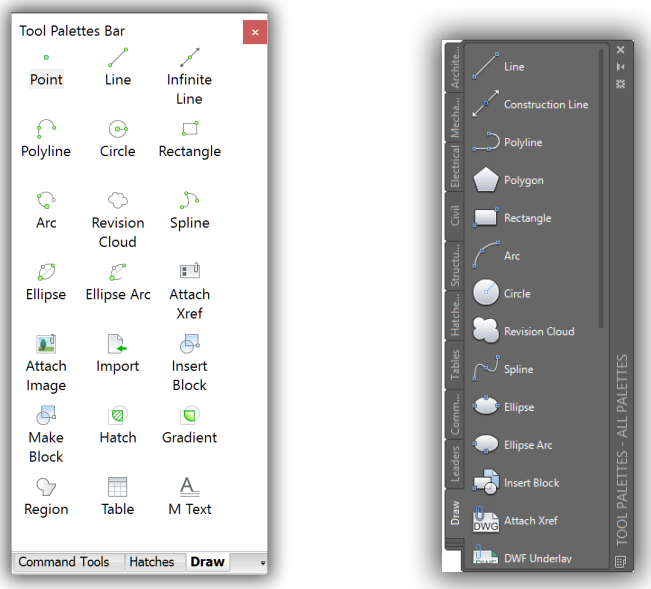


Left: *Properties bar in BricsCAD*
Right*: Properties palette in AutoCAD*

(NEW IN V16) **Transparency**, **Clipping**, **Front Plane**, **Back Plane**, and **Default Lighting** are new properties added to the Properties bar.

DIFFERENCES IN TOOL PALETTES

Tool palettes operate similarly in both CAD programs, except that BricsCAD customizes them differently. See chapter 4 for a tutorial on customizing the Tool Palettes bar.



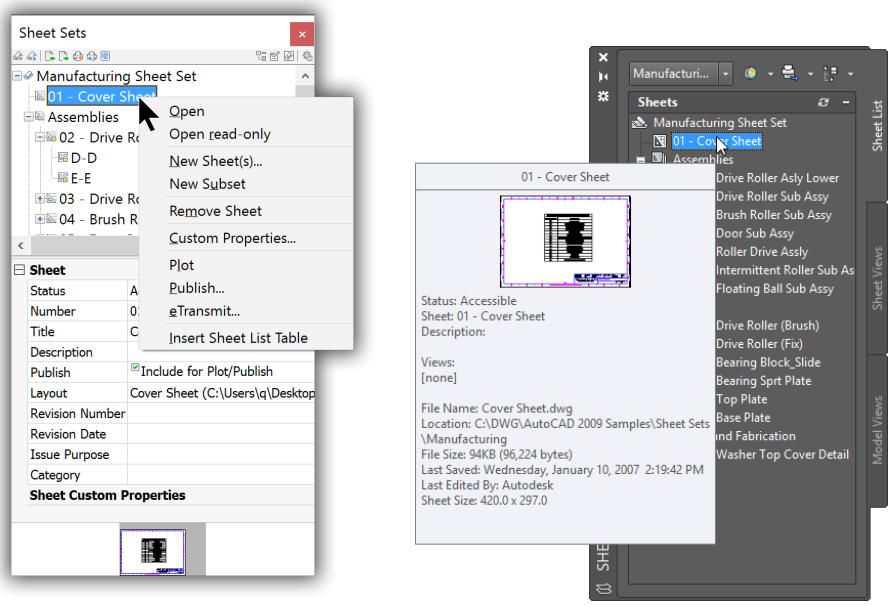
Left: *Tool Palette bar in BricsCAD*
Right: *Tool palette in AutoCAD*

One other difference: whereas AutoCAD stores tool palette definitions in ATP files (short for "AutoCAD tool palettes"), BricsCAD stores them in BTP files (short for "BricsCAD tool palettes"). Both file types use XML as the format.

DIFFERENCES IN SHEET SETS

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

(NEW TO V16) BricsCAD's user interface looks like AutoCAD's, a single palette. (Prior to V16 BricsCAD displayed sheetset data in the "Drawing Explorer.") Both CAD programs use right-click menus and a toolbar to create, edit, and publish sheet sets.



Above: *BricsCAD's sheetset manager*
Below: *AutoCAD's sheetset manager*

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:

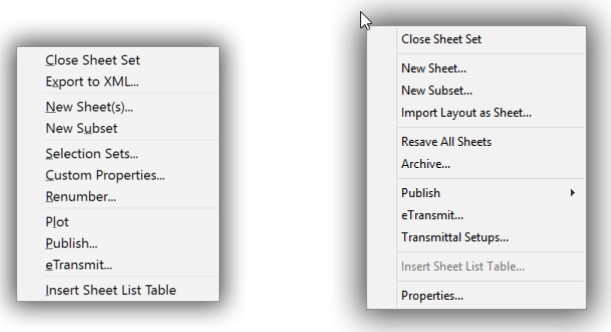
Left: *Sheet set toolbar in BricsCAD*
Right: *Sheet set toolbar in AutoCAD*

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 (NEW TO V16)
- 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*

(NEW TO V16) **Insert Sheet List Table** is added to BricsCAD's shortcut menu.

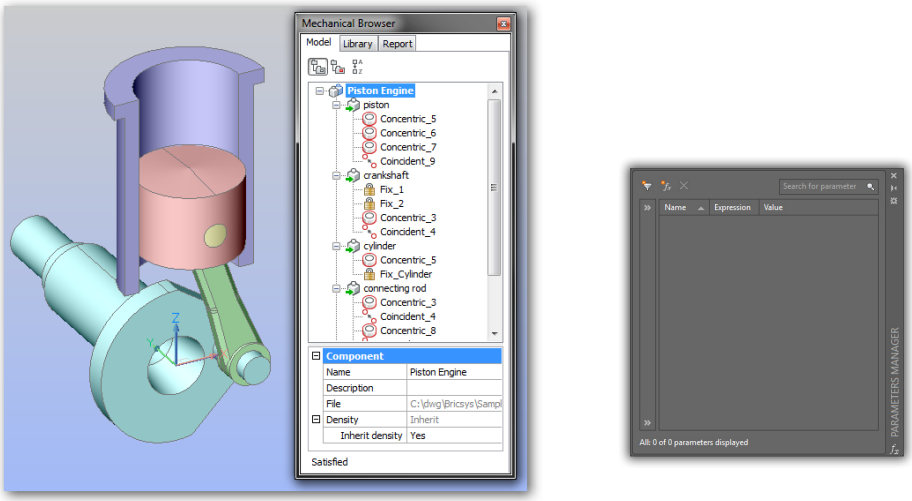
BricsCAD uses the same DST format as AutoCAD's sheet sets, and so you can reuse them from AutoCAD. In addition, BricsCAD can import and export sheet set files in XML format.

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	...
3D dimensional constraints	3	...
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.)



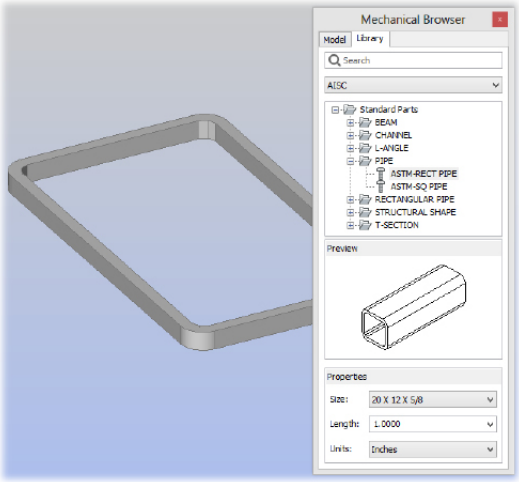
Left: *BricsCAD's Mechanical Browser handles constraints, parameters, and assembly parts*
Right: *AutoCAD's Parametrics Manager handles constraint formulas only*

(NEW TO V16) The **Report** tab is added to the Mechanical Browser bar.

Constraints added to models in BrisCAD are not recognized in AutoCAD. BricsCAD, however, reads constraints in 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.

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. (AutoCAD has a parts library named Content Browser, but it does not include parts.)



Parts library in BricsCAD

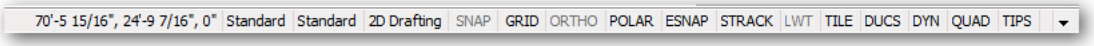
Status Bar & Other UI Differences

Here you get overviews of the differences in other user interface elements: status bar, working sets (in 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, but the two have some differences in the functions they provide. BricsCAD continues to use text for the buttons, while AutoCAD allows users to view icons only, which can be confusing.

Above: *The default status bar in AutoCAD*
Below: *The default status bar in BricsCAD*

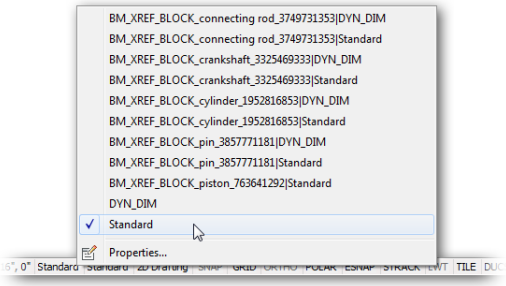


This is the list of similarities and differences of the contents of the two status bars:

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	Yes	Yes	
Grid	Yes	Yes	
Infer Constraints	Yes	...	BricsCAD has design intent
Isometric Drafting	Yes	...	BricsCAD has isometric mode
Polar	Yes	Yes	
2D OSnap/Esnap	Yes	Yes	
3D OSnap	Yes	...	BricsCAD has 3D entity snapping
OTrack / STrack	Yes	Yes	
LWT	Yes	Yes	
Transparency	Yes	...	BricsCAD sets transparency through variables
Selection Cycling	Yes	...	BricsCAD cycles through selections
Model / Tile	Yes	Yes	
DUCS	Yes	Yes	
Selection Filtering	Yes	...	BricsCAD has not have selection cycling
Gizmo	Yes	...	BricsCAD does not have the 3D editing gizmo
Tips	...	Yes	
Annotation Visibility	Yes	...	

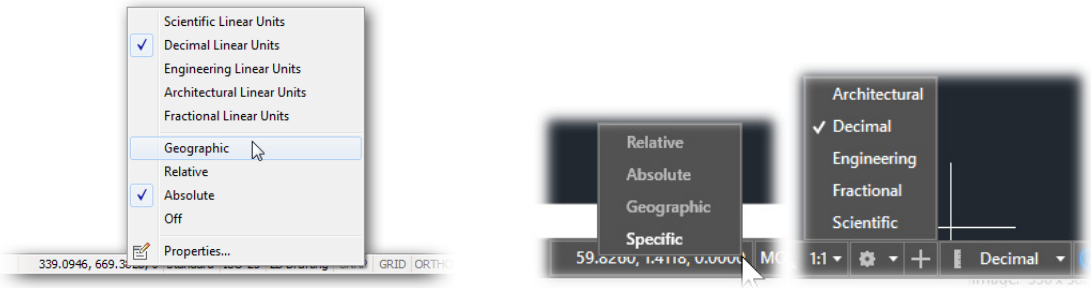
Status Bar Function	AutoCAD	BricsCAD	Notes
AutoScale	Yes	...	
Annotation Scale	Yes	Yes	
DYN	Yes	Yes	Dynamic input toggle
QUAD	...	Yes	Quad cursor toggle
Tablet	...	Yes	
Units	Yes	(Yes)	BricsCAD settings is in Coordinates shortcut menu
Quick Properties	Yes	...	BricsCAD does not have Quick Properties
Graphics 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.)



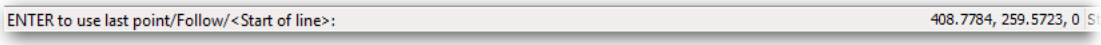
Accessing dimensions styles from the status bar in BricsCAD

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.)



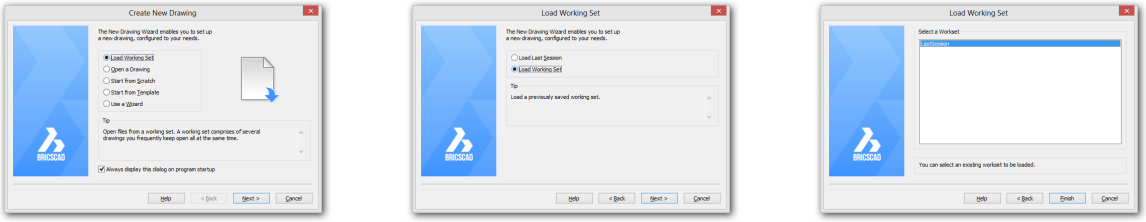
Status bar in BricsCAD displaying command prompts

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 possible 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 one 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.

The Create New Drawing dialog box includes Load Working Set as one of its options. After BricsCAD opens, you access other worksets through the **Workset** command, or by choosing **File** on the menu bar and then selecting **Working Sets**.

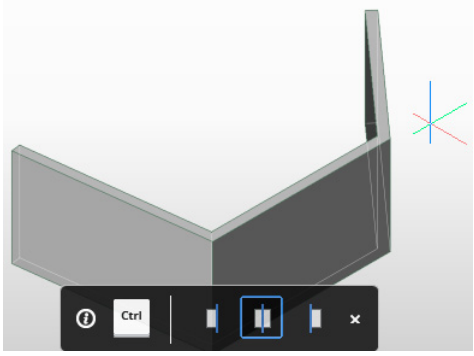


Left to right: *Steps in loading a working set upon starting up BricsCAD*

TIPS WIDGET (BRICSCAD ONLY)

"Tips" are like interactive tooltips. (AutoCAD ha 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 display can be toggled through the TIPS button on the status bar.

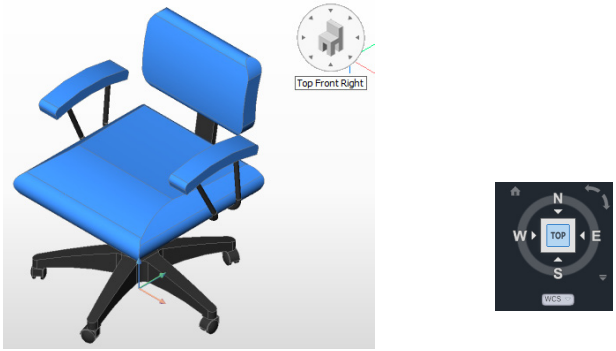
The Tips widget shows several icons. They indicate that by holding down the **Ctrl** key during the command, the user can change the justification between left, centered, and right. Pause the cursor over the Tip to get a brief description of the purpose. Click the **x** to dismiss the Tip.



The Tips widget for the Polysolid command

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.



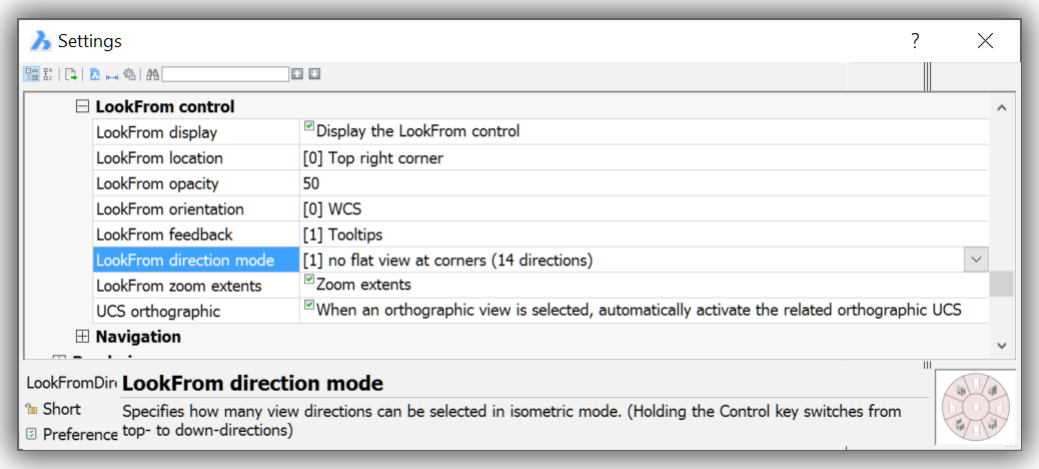
Left: *LookFrom control in BricsCAD*
Right: *ViewCube control in AutoCAD*

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>:
```

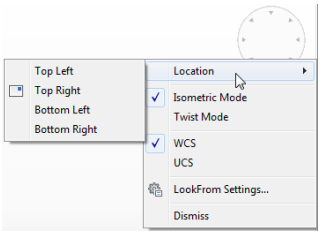
Turn it off for 2D drafting. The Settings option opens the Settings dialog box at the LookFrom section.

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:

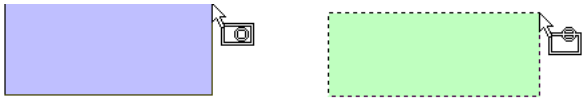
Left: *Isometric mode*
Right: *Twist mode*

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:

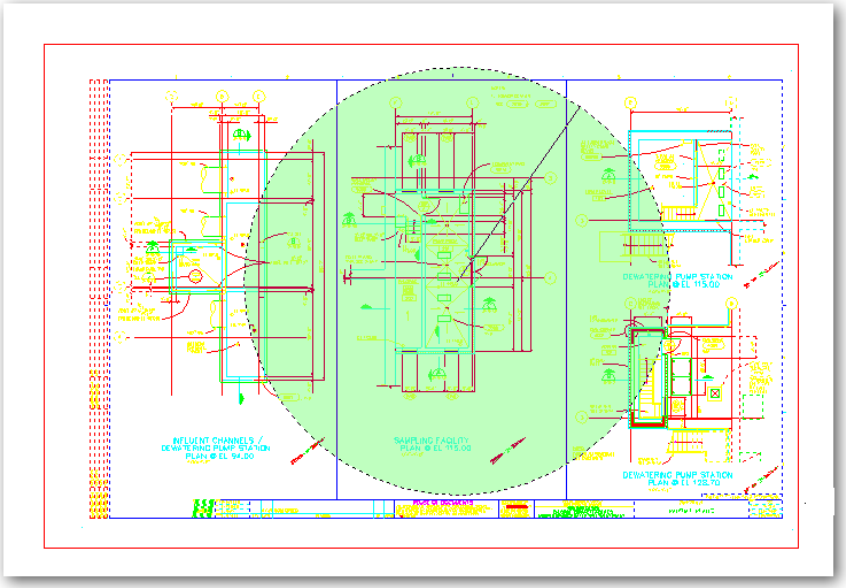
```
: select
Select entities to include in set: ?
Select entities:  ALL/Add/+/Remove/-/Previous/Last/Window/Crossing/Outside/WPolygon/CPoly-
gon/OPolygon/WCircle/CCircle/OCircle/Box/POint/Fence/AUto/Multiple/Single/PROperties/Dialog/
Undo/Group:
```

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

- **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.



BricsCAD selecting all objects inside a circular selection window

VARIATIONS IN DESIGNCENTER & DRAWING EXPLORER

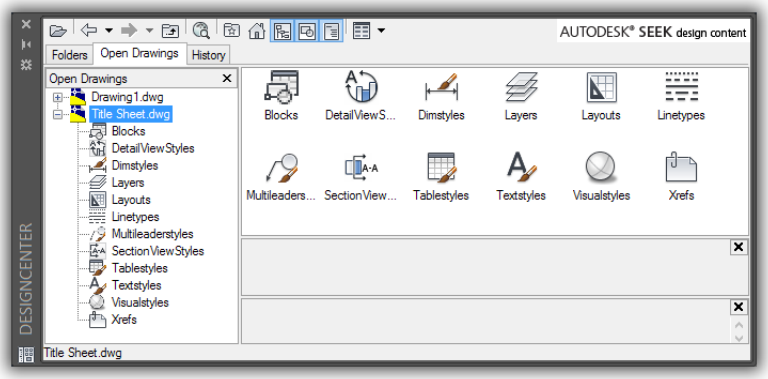
BricsCAD's Drawing Explorer is best compared with AutoCAD's DesignCenter, but Explorer reports more information and provides greater control over drawing elements. Drawing Explorer centralizes in BricsCAD what in AutoCAD amounts to as many separate dialog boxes; facilities such as layer management, UCS 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.

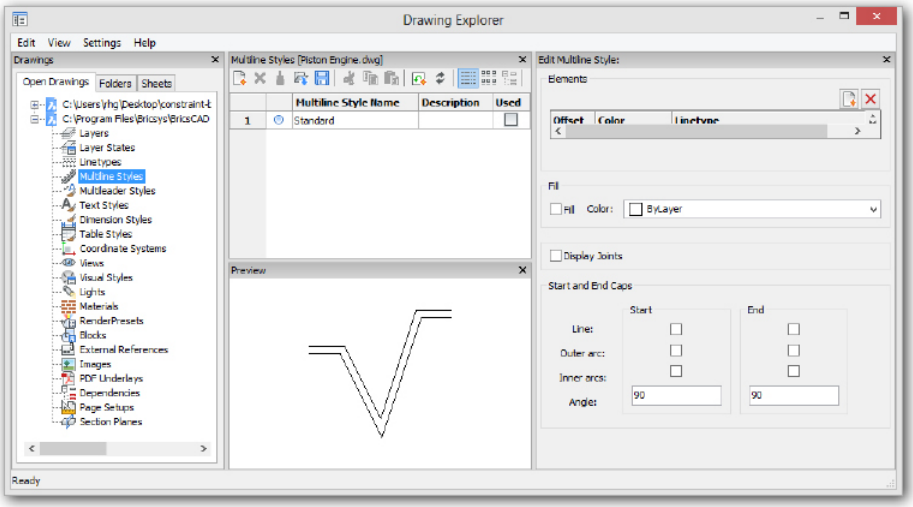
AutoCAD's DesignCenter Node	**BricsCAD's Drawing Explorer Node**	**AutoCAD Commands** for functions not in DesignCenter
Blocks	Blocks	Block and Bedit commands
...	Chapoo (in Folders tab)	SaveToCloud command (Autodesk 360)
...	Coordinate Systems	UcsMan command
...	Dependencies	eTransmit command
DetailViewStyles	...	ViewDetailStyle command
Dimstyles	Dimension Styles	DimStyle command
Xrefs	Xrefs	ExternalReferences command
...	Images	ExternalReferences command
Layers	Layers	Layers command
...	Layer States	LayerStates command
...	Lights	LightList command
Linetypes	Linetypes	-Linetype command
...	Materials	MatBrowserOpen command
Layouts	Page Setups	PageSetup command
...	Multiline Styles	MlStyle command
Multileaderstyles	Multileader Styles	MultileaderStyle command
...	PDF Underlays	PDFAttach command
...	Render Presets	RenderPresets command
...	Section Planes	SectionPlaneSettings command
SectionViewStyles	...	ViewSectionStyle command
Tablestyles	Table Styles	TableStyle command
Textstyles	Text Styles	Style command
...	Views	View command
VisualStyles	Visual Styles	VisualStyles command

(NEW IN V16) The Sheet Sets node is moved from Drawing Explorer to its own bar.

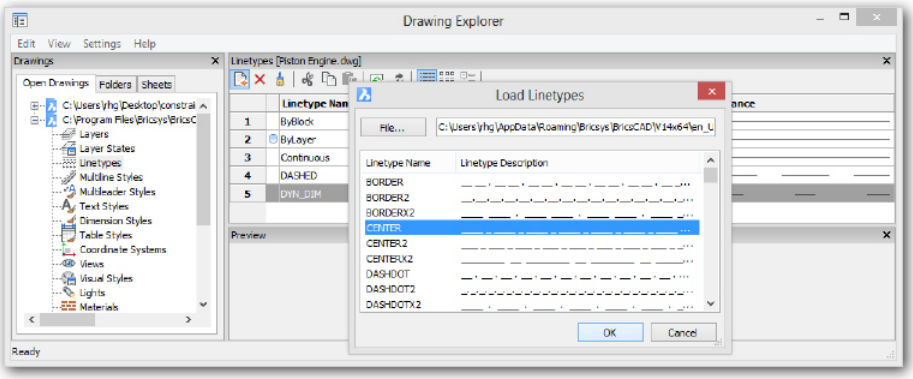
To access BricsCAD's Drawing Explorer, enter the **Explorer** command or from the **Tools** menu: chose **Drawing Explorer**. BricsCAD displays Drawing Explorer automatically when you enter related commands, such as Layer and Xref.



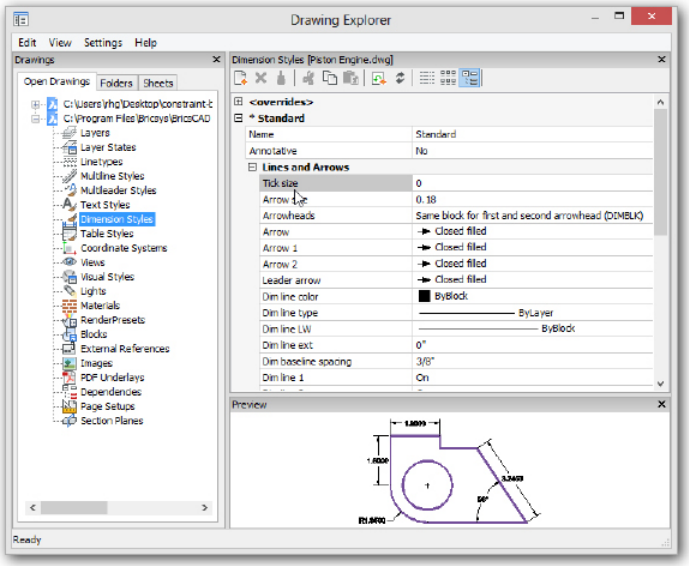
Above: *AutoCAD's Design Center.*
Below: *BricsCAD's Drawing Explorer.*



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

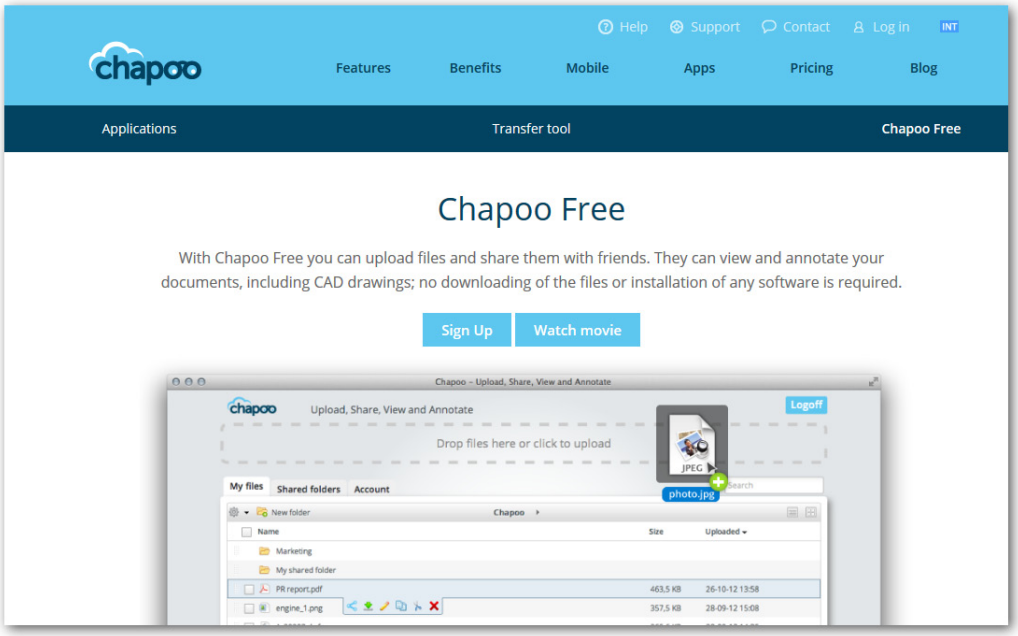
Chapoo is the online collaboration and cloud storage from Bricsys. The equivalent in AutoCAD is Autodesk 360. Commands inside BrisCAD let you open and save files from and to the Chapoo cloud.

Inside Chapoo, you can 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

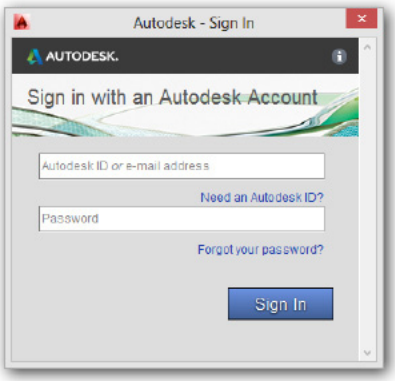
To sign up for the free version of Chapoo, go to http://chapoo.com/en_INTL/apps/free.



Accessing Chapoo for the first time

Using Chapoo

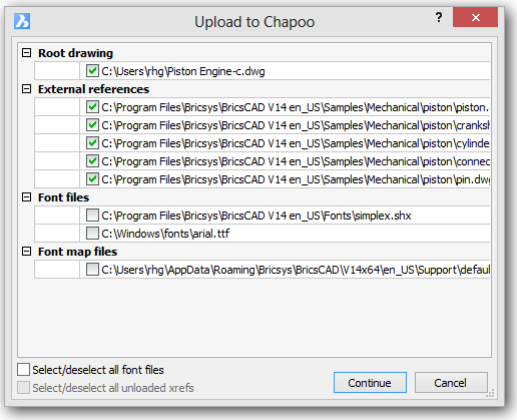
To log into Chapoo from BricsCAD, enter the **ChapooOpen** command or from the **File** menu choose **Chapoo | Open**.



Left: *Logging into Chapoo from BricsCAD*
Right: *Logging into 360 from AutoCAD*

Commands in BricsCAD let you upload and download files:

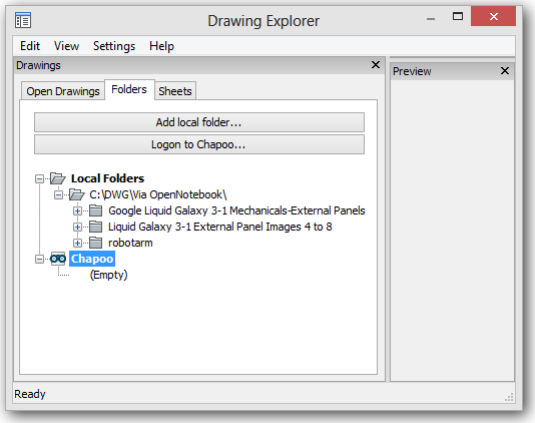
ChapooOpen opens files stored online
ChapooDownload downloads files from online to your computer
ChapooUpload uploads the current file to your online account, along with all dependent files, such as xrefs and image files, and optionally uploads fonts



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 2016. 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 V16	AutoCAD 2016
Oldest DWG format	Release 12 (from 1993)	Release 14 (from 1997)
Oldest DXF format	Release 9 (1987)	Release 12 (1993)
Newest DWG/DXF format	Release 2013-6	Release 2013-6

In summary, BricsCAD V16 reads and writes all the same DWG and DXF files as AutoCAD 2016 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 2016. 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 2013-6 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 2013-6 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 2016 ENTITIES

The following checklist shows the names all entities supported by DWG 2016. Those with 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 2013-6 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 2016, 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 V16.
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 V16 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. The objects can be edited only 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 "true" point-based 3D mesh objects introduced to AutoCAD 2010, and not the "old" meshes made from polyfaces. BricsCAD also 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.

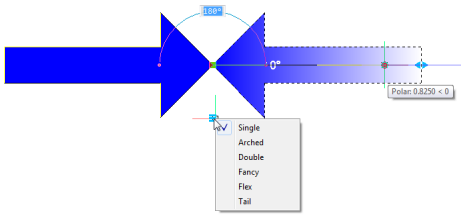
Inspection Dimensions. BricsCAD displays inspection dimensions made by AutoCAD's DimInspect command, but cannot edit or create them.

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 in the United States against Autodesk (AutoCAD), Dassault Systemes (SolidWorks), and Simens PLM Software (Solid Edge). The quick dimensioning capabilities were allegedly patented by Adra Systems. Siemens PLM has settled by making a payment to the patent troll; Autodesk and Dassault continue to defend their implementations of QDim. 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

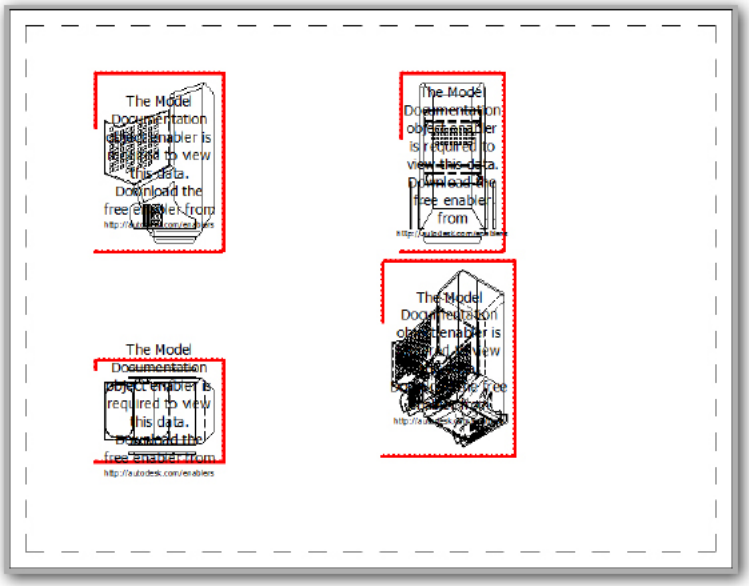
BricsCAD specifies geographic locations with its GeographicLocation command, but does not display, create, or edit the marker glyphs that mark locations placed in AutoCAD.

Layers

BricsCAD can read, edit, and write layers and layer states, but can't apply all the formatting AutoCAD can.

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

Instead, BricsCAD has its own form of model documentation called view generation. It creates 2D plan and isometric views of 3D models, and makes sections and detail views of them.

Multilines

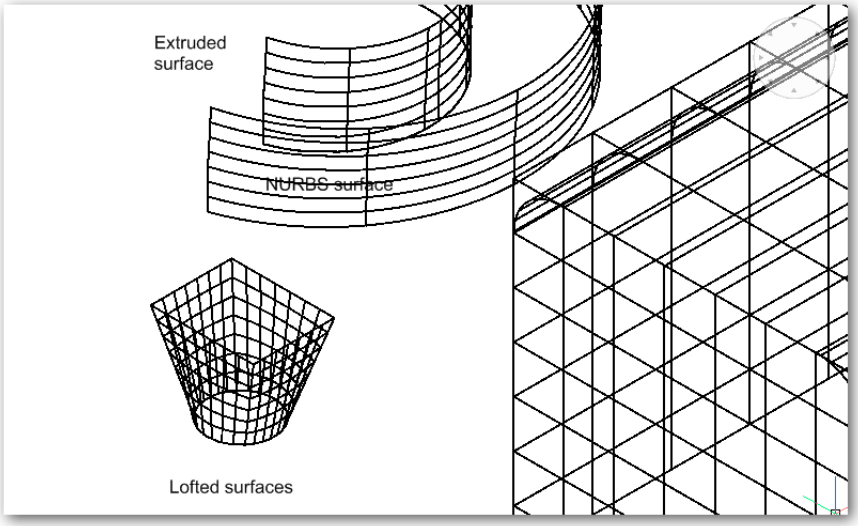
BricsCAD reads and creates multilines and multiline styles with AutoCAD's MLine and MlStyle commands. BricsCAD, however, lacks the MlEdit command, and so intersections (vertices) cannot be edited. Some aspects of multilines can be edited with grips and through the Properties bar's option. The BricsCAD 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 the AutoCAD Architecture one provided by Open Design Alliance. BricsCAD edits only the basic properties of proxy objects (color, linetype, and so on) through the Properties bar.

Surfaces

(NEW TO V16) BricsCAD creates 3D surfaces just AutoCAD does: if a command like Extrude or Revolve works with an open object, like a line or an arc, then the result is a 3D surface; if a closed object, like a polygon or circle, then a 3D solid. BricsCAD recognizes all surfaces created by AutoCAD, including NURBS and swept surfaces.



Surfaces created in AutoCAD and displayed by BricsCAD

Being a new entity to BricsCAD, functions are missing. For example, the Loft command creates only 3D solids; 3D surface lofts have to wait for a future release. In the meantime, BricsCAD recognizes surface lofts created in AutoCAD.

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 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
- Details and sections in 2D views generated from 3D models

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

DWG 2013-6 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 V16 supports DWG v20.0, which includes entities generated by 2016 and earlier; Autodesk added no new entities to AutoCAD 2016.

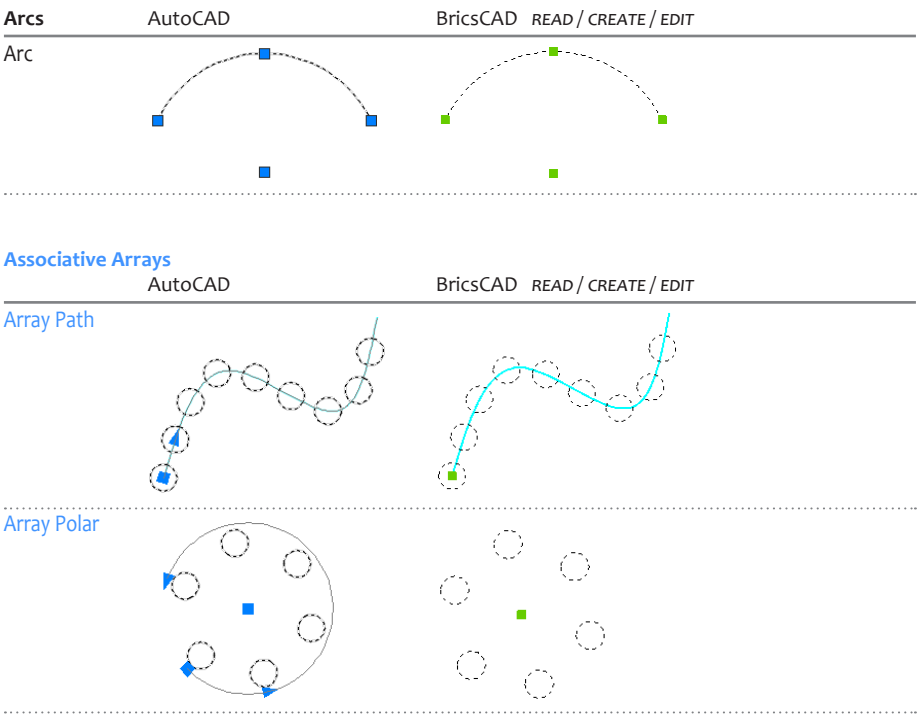
Entities are listed in alphabetical order under the following sections.

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

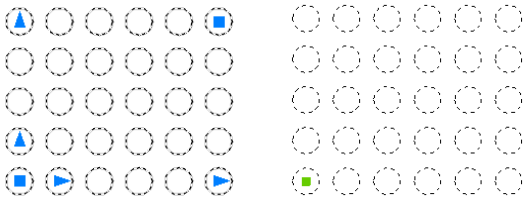
Equivalent entities are illustrated from AutoCAD and BricsCAD, with entity grips shown.

2D ENTITIES

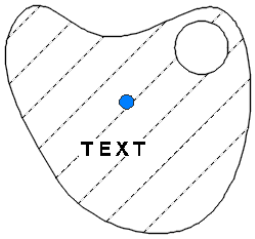
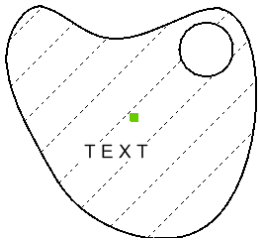
BricsCAD accurately displays the following 2D entities created in AutoCAD 2016:

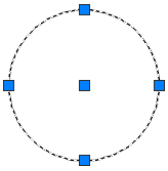
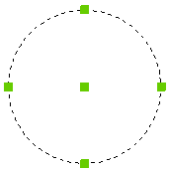


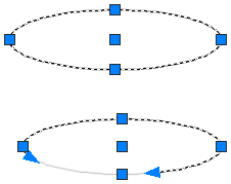
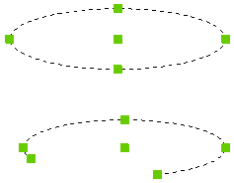
Array Rectangular

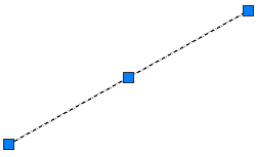
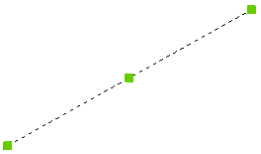


Associative Hatches

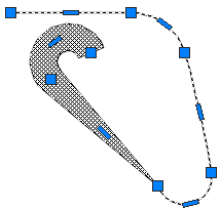
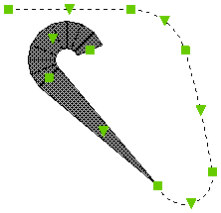
	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Hatch		

Circles	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Circle		

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

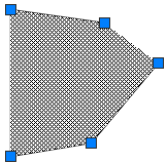
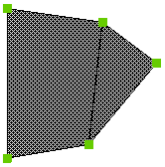
Lines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Line		

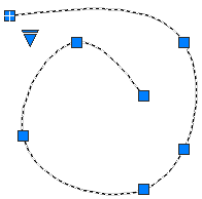
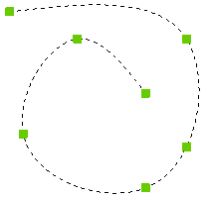
Points	AutoCAD	BricsCAD *READ / CREATE / EDIT*
PdMode 95		

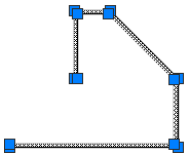
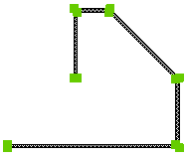
Polylines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Polyline		

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 2016. The exceptions are some formatting of mtext and tables, and annotative scaling, as detailed later in this chapter.

Attribute Definitions

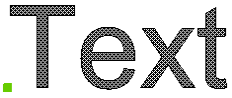
	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Attribute Definition		

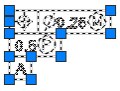
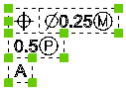
Attribute References

	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Attribute Reference		

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

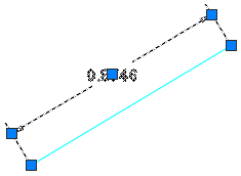
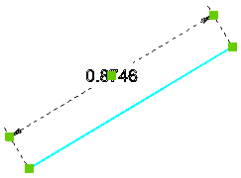
*) BricsCAD does not create all aspects of mtext; see "Compatibility of Styles" later in the chapter.

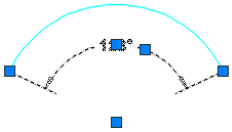
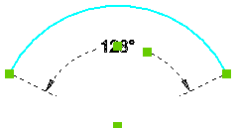
Text	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Text		

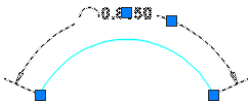
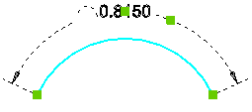
Tolerances	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Tolerance		

DIMENSION ENTITIES

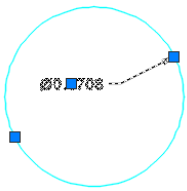
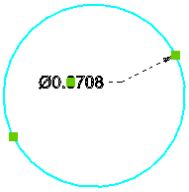
BricsCAD supports all aspects of AutoCAD's dimension entities, except that it cannot create or edit broken, inspection, and jogged dimensions.

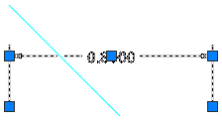
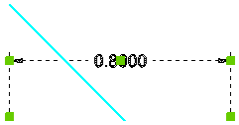
Aligned	**AutoCAD**	**BricsCAD** *READ / CREATE / EDIT*
Aligned Dimension		

Angular	**AutoCAD**	**BricsCAD *** *READ / — / —*
Angular Dimension		

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

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

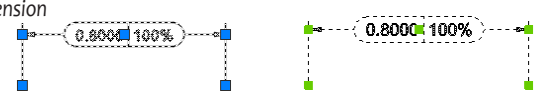
Diameter	**AutoCAD**	**BricsCAD** *READ / CREATE / EDIT*
Diametric Dimension		

DimBreak	**AutoCAD**	**BricsCAD *** *READ / — / —*
Broken Dimension Line		

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

DimInspect **AutoCAD** **BricsCAD *** *READ / — / —*

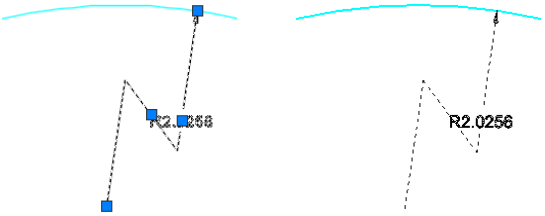
Inspection Dimension



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

DimJogged **AutoCAD** **BricsCAD *** *READ / — / EDIT*

Jogged Dimension



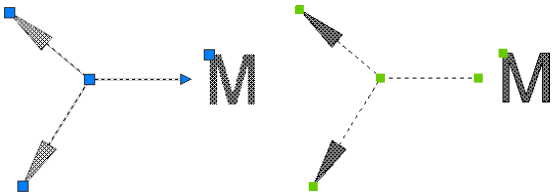
*) BricsCAD displays and edits jogged dimensions, but does not create them.

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

Leader

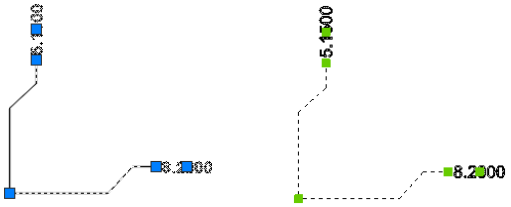
Multileaders **AutoCAD** **BricsCAD** *READ / — / —*

Multiline Leader



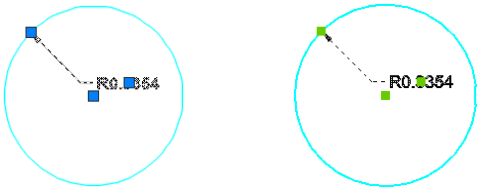
Ordinate **AutoCAD** **BricsCAD** *READ / CREATE / EDIT*

Ordinate



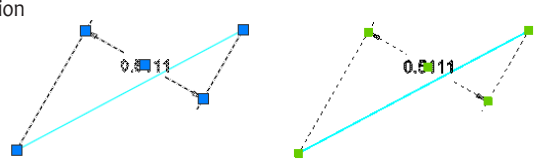
Radius **AutoCAD** **BricsCAD** *READ / CREATE / EDIT*

Radial Dimension



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

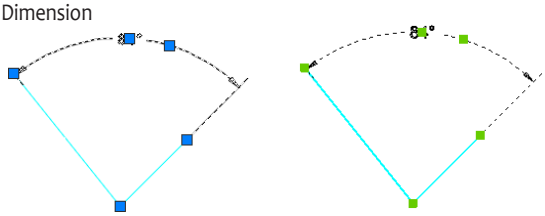
Rotated Dimension



Three-Point Angular

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

3-point Angular Dimension

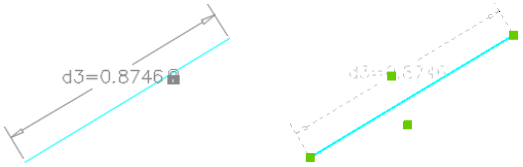


GEOMETRIC AND DIMENSIONAL CONSTRAINTS

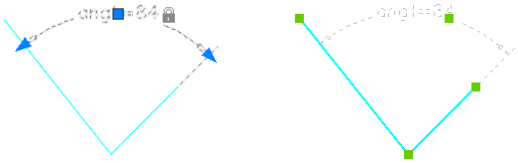
Dimensional Constraints

AutoCAD BricsCAD *READ / CREATE / EDIT*

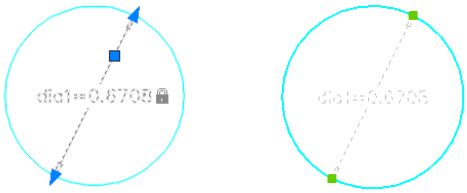
Aligned Dimensional Constraint



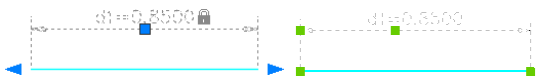
Angular Dimensional Constraint



Diameter Dimensional Constraint

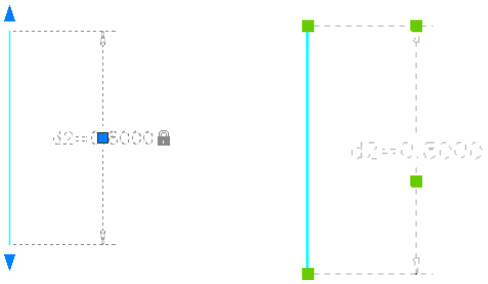


Horizontal Dimensional Constraint

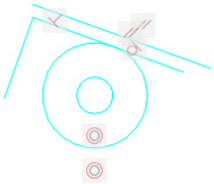
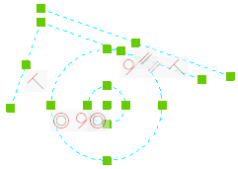


Radius Dimensional Constraint

Vertical Dimensional Constraint

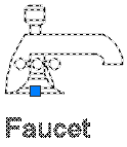
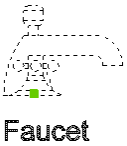
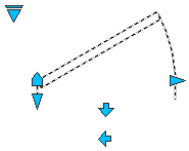
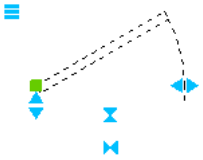


Geometric Constraints

	AutoCAD	**BricsCAD** *READ / CREATE / EDIT*
Geometric constraint		

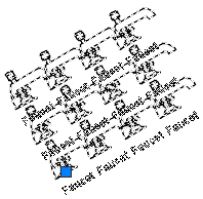
COMPLEX 2D ENTITIES

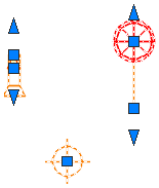
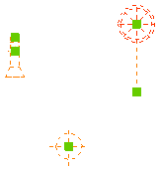
Block References

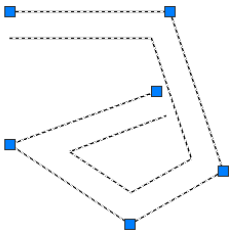
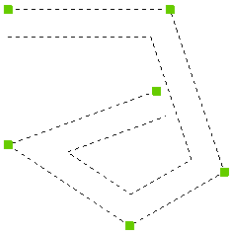
	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Insert	 Faucet	 Faucet
Dynamic *		

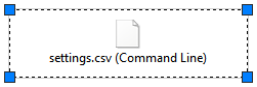
*) BricsCAD displays and edits dynamic blocks, but does not create them.

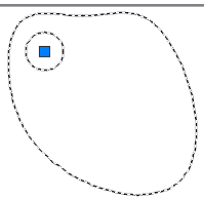
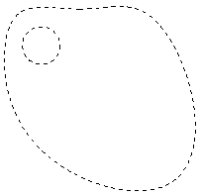
MInsert

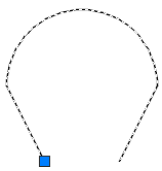
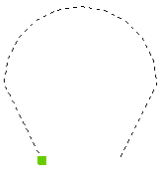


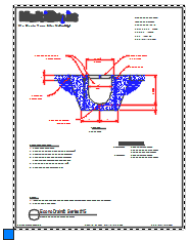
Lights	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Light		

Multilines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Mline		

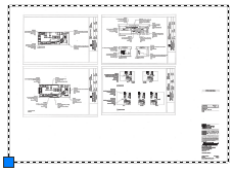
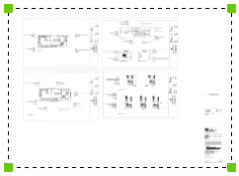
OLE Frames	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Ole		

Regions	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Region		

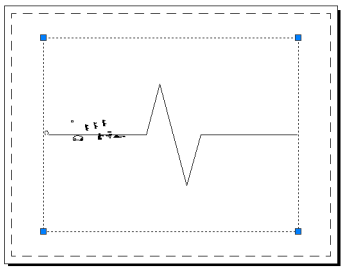
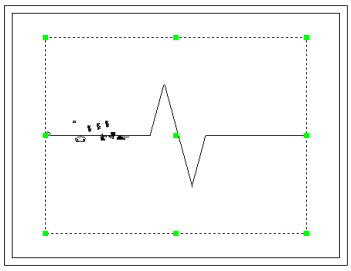
Shapes	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Shape		

Underlays	AutoCAD	BricsCAD * *PARTIAL READ / CREATE / EDIT*
DWF Underlay		

*) BricsCAD does not display DWF underlays.

PDF Underlay		

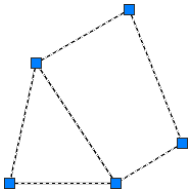
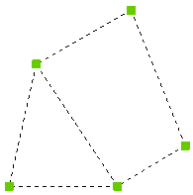
*) Bricsys does not support multi-page PDF files.

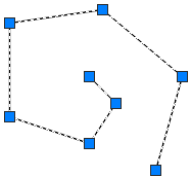
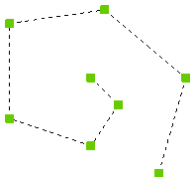
Viewports	AutoCAD	BricsCAD * *READ / CREATE / PARTIAL EDIT*
Viewport		

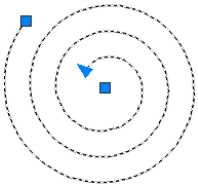
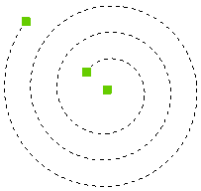
*) BricsCAD does not invert clipped viewports.

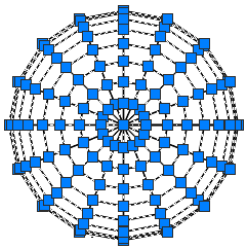
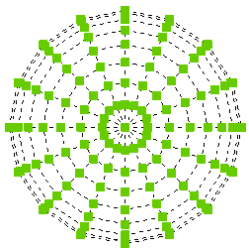
3D ENTITIES

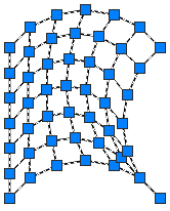
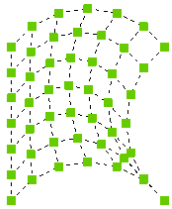
BricsCAD accurately displays the following 3D entities created in AutoCAD 2016:

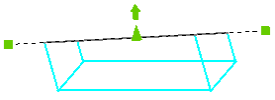
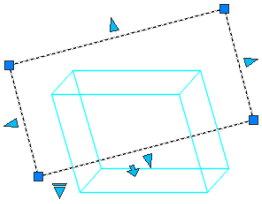
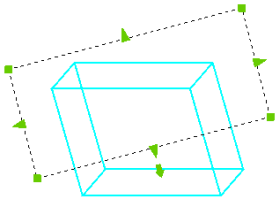
3D Faces	AutoCAD	BricsCAD *READ / CREATE / EDIT*
3D face		

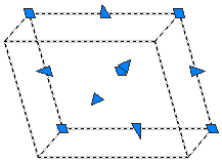
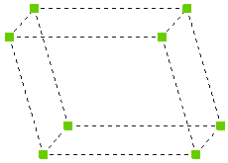
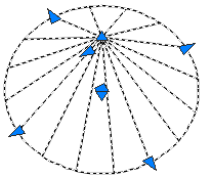
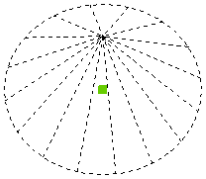
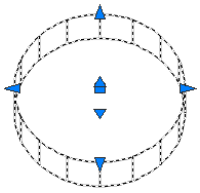
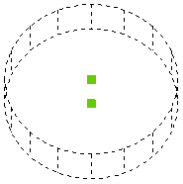
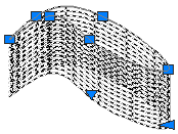
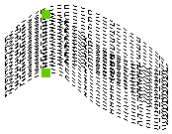
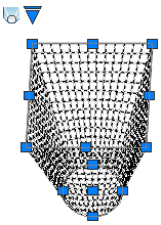
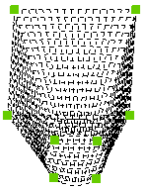
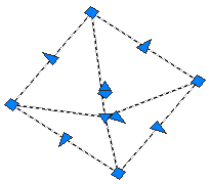
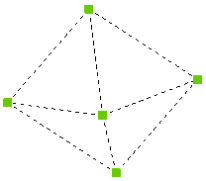
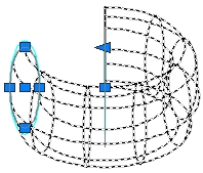
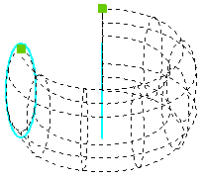
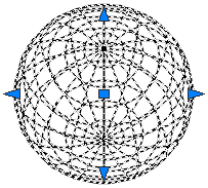
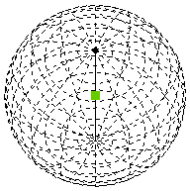
3D Polylines	AutoCAD	BricsCAD *READ / CREATE / EDIT*
3D polyline		

Helixes	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Helix		

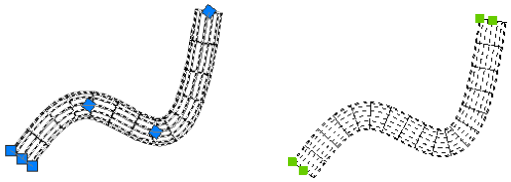
Polyface Meshes	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Polyface Mesh		

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

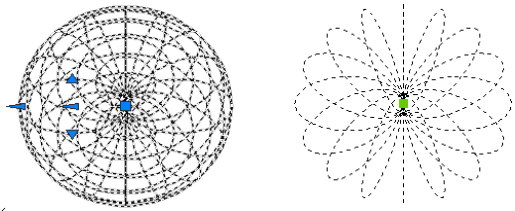
Sections	AutoCAD	BricsCAD *READ / CREATE / EDIT*
Section Line		
Section Object		

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

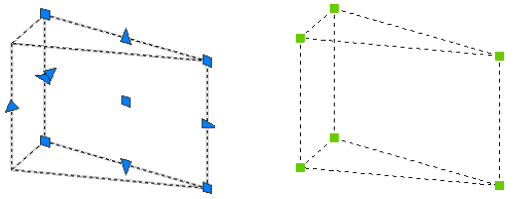
Sweep



Torus

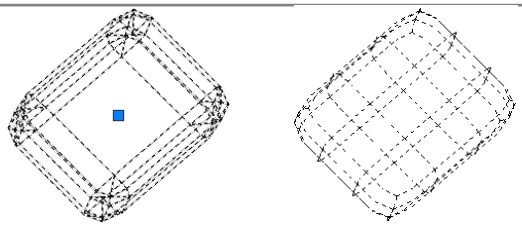


Wedge



Subdivisions	AutoCAD	BricsCAD *	*READ* / — / —

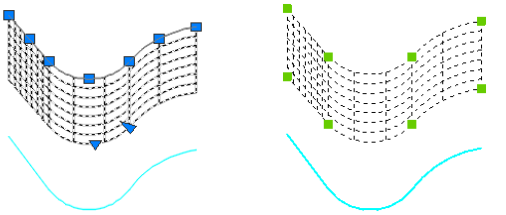
Mesh



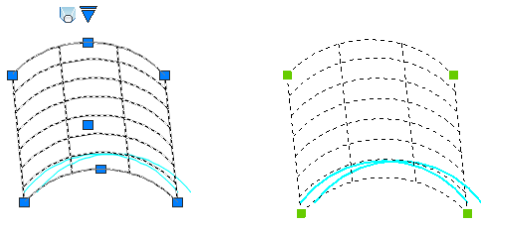
*) BricsCAD recognizes mesh objects created by AutoCAD, but cannot create or manipulate 3D meshes. The objects can be edited using basic commands (such as Move, Copy, and Delete), and their basic properties can be modified, such as color and linetype. (Note that these are the "true" 3D mesh objects introduced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

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

Extrusion

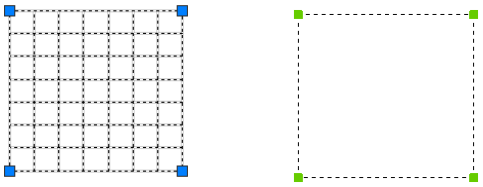


Loft (read and edit only)

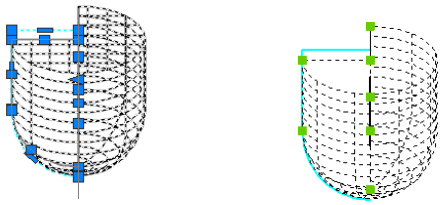


NURBS

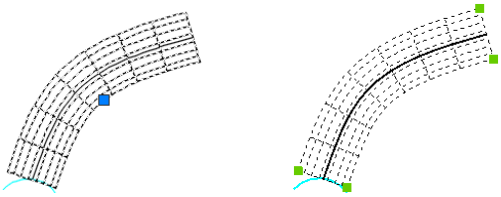
Planar



Revolve



Sweep



*) Surface creation is new to BricsCAD V16, and it reads all surfaces created in AutoCAD. BricsCAD does not, however, create lofted surfaces at time of writing.

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, but not properties listed later
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	...	
Thickness	Thickness	
Transparency	Transparency	

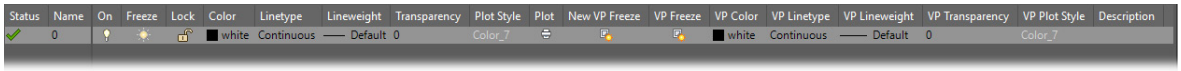
LAYER PROPERTY COMPATIBILITY

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.

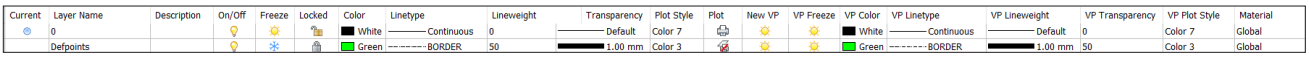
BricsCAD supports layer states and filters. Transparency and per-viewport settings are missing from BricsCAD; on the other hand, BricsCAD supports the Material property in directly layers, whereas AutoCAD does only indirectly.

AutoCAD Command	BricsCAD Command
Layer	Layer
LayerState	LayerState
LayerP	LayerP

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



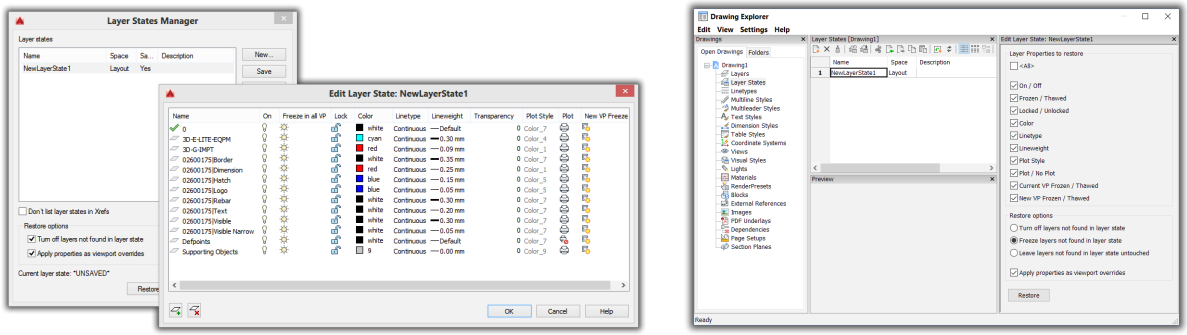
Above: *Layer properties in AutoCAD (paper space)*
Below: *Layer properties in BricsCAD (paper space)*



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 ll 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. (NEW TO V16) The Layer States manager is redesigned in V16.



Left: *Layer States Manager dialog boxes in AutoCAD*
Right: *Layer States in BricsCAD's Drawing Explorer; the Layer Properties to Restore panel is new in V16*

Compatibility Between Styles

BricsCAD supports most of the styles found AutoCAD. Those shown in blue are since the last edition of this book.

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.

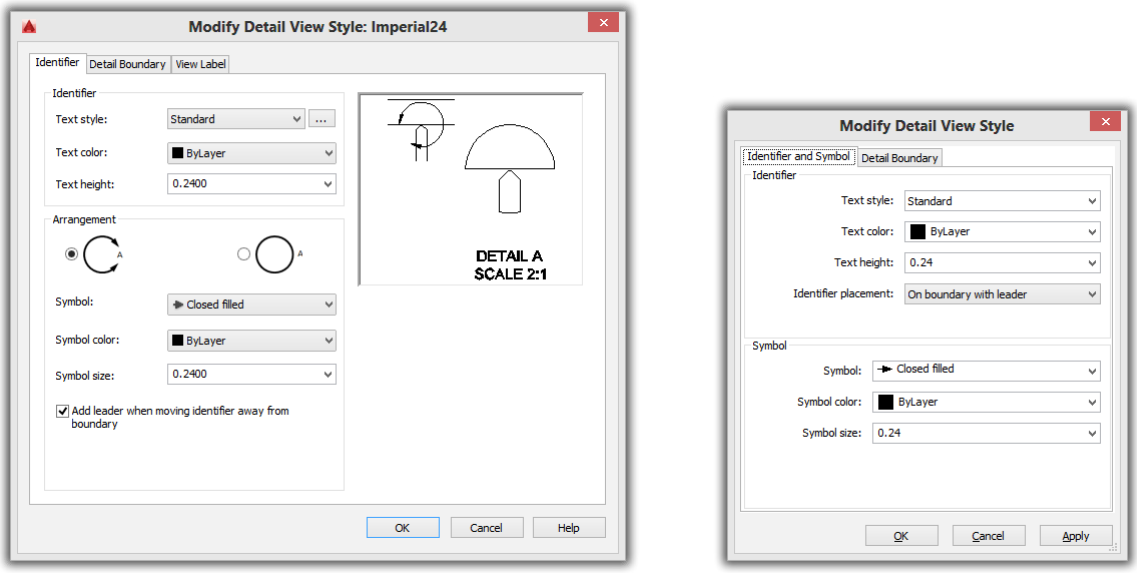
View Detail and Section Styles

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.

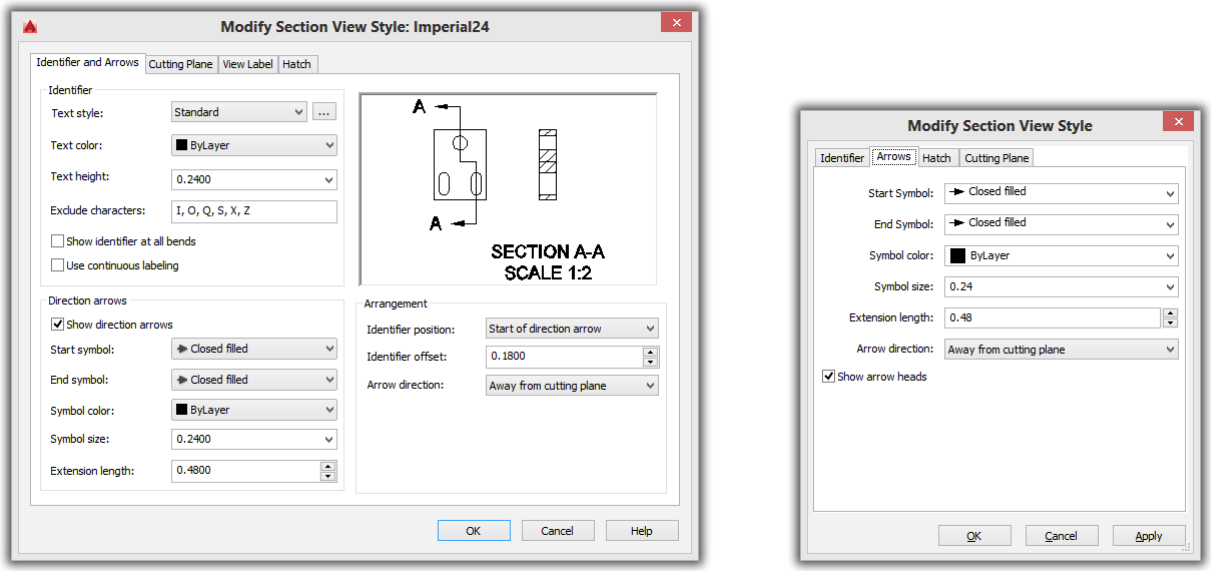
AutoCAD Commands	BricsCAD Commands
ViewDetailStyle	ViewDetailStyle
ViewSectionStyle	ViewSectionStyle

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:



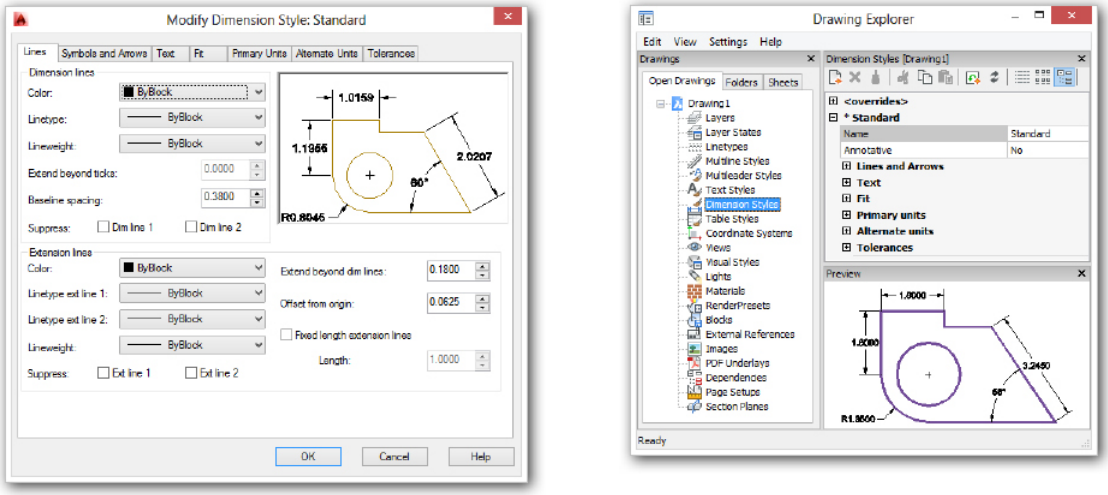
Left: *AutoCAD's tabbed Modify Section View Style dialog box;* **right:** *BricsCAD's tabbed Modify Section View Style dialog box*

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:



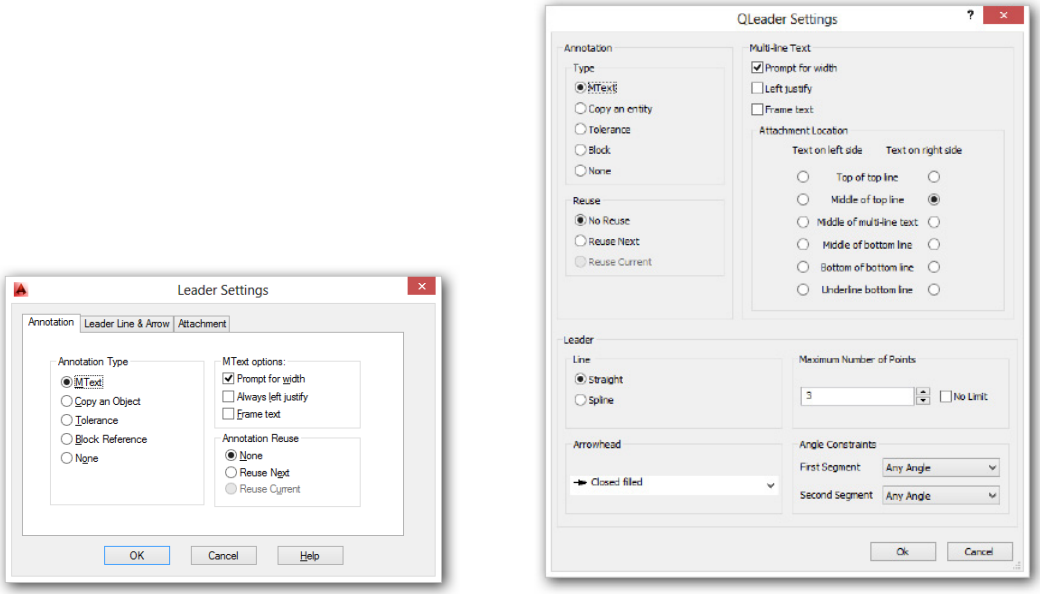
Left: *AutoCAD's DimStyle tabbed dialog box;* **right:** *BricsCAD's Drawing Explorer for dimensions styles*

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.

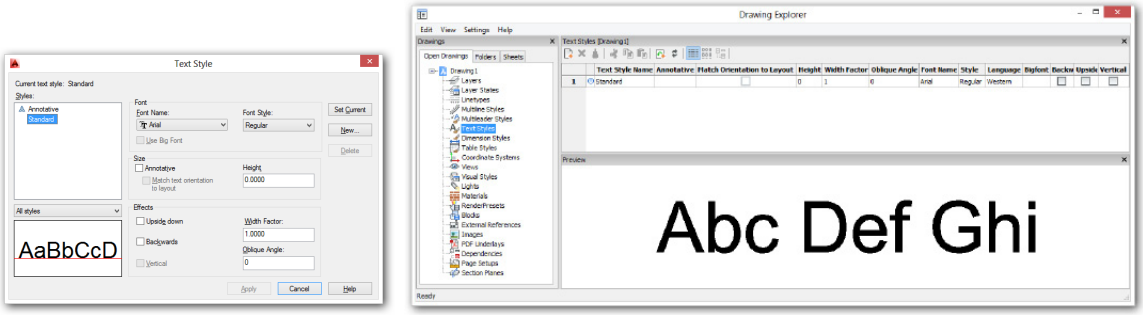
AutoCAD QLeader Option	Equivalent BricsCAD Option
Annotation options	
Annotation Type	Type
MText Options	Multi-line Text
Annotation Reuse	Reuse
Leader Line & Arrow options	
Leader Line	Leader
Number of Points	Maximum Number of Points
Arrowhead	Arrowhead
Angle Constraints	Angle Constraints
Attachment 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:

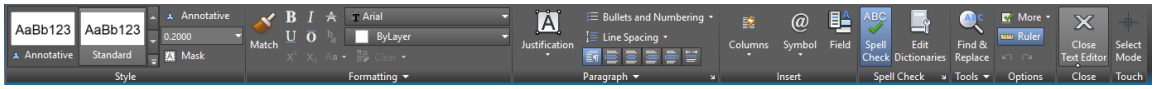
Left: *Annotation icon used by AutoCAD;* ***right:*** *As employed by BricsCAD*

AutoCAD Command	BricsCAD Command
Style	Style
MText	MText

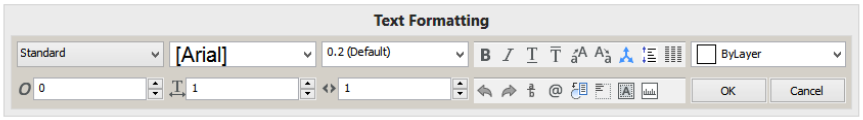


Left: *AutoCAD's Style dialog box;* **right:** *BricsCAD's Drawing Explorer for text styles.*

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.

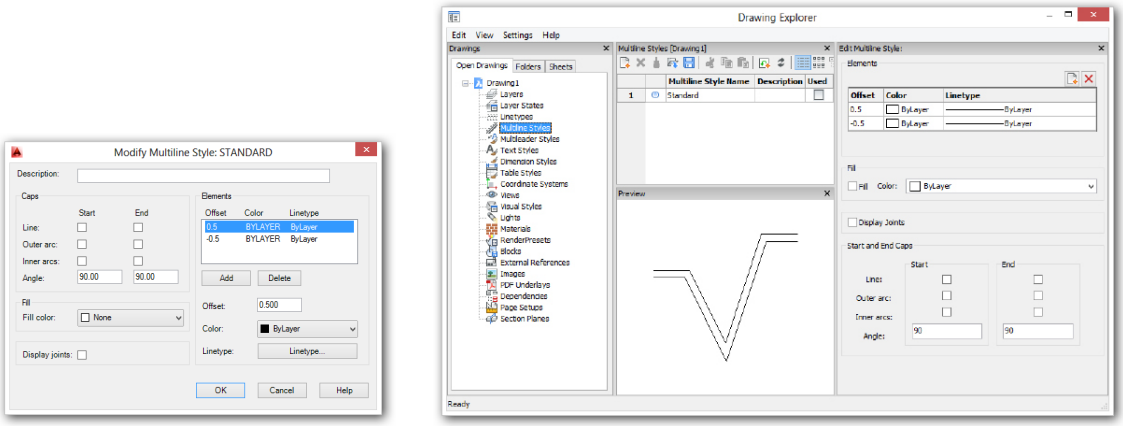
AutoCAD Mtext Function	BricsCAD Mtext Function
Style	Style
Font	Font
Annotative	Annotative
Height	Height
Boldface	Boldface
Italicized	Italicized
Underline	Underline
Overline	Overline
Undo	Undo
Redo	Redo
Fractions	Fractions
Color	Color
Ruler Toggle	Ruler Toggle
Dynamic or Static Columns	Dynamic or Static 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.



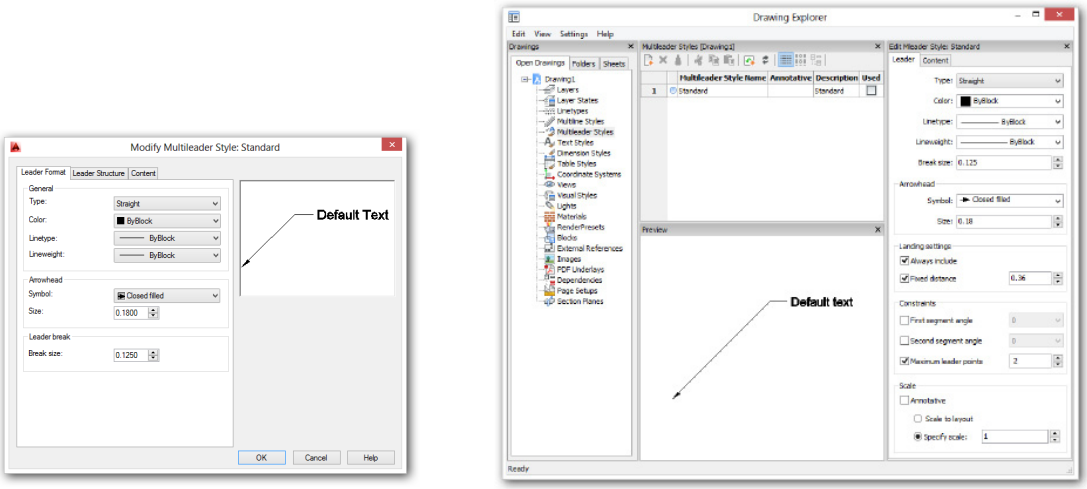
Left: *AutoCAD's multiline style editor;* ***right*** *BricsCAD's multiline style editor in Drawing Explorer*

Multiline Leader Styles

BricsCAD creates multilines 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.



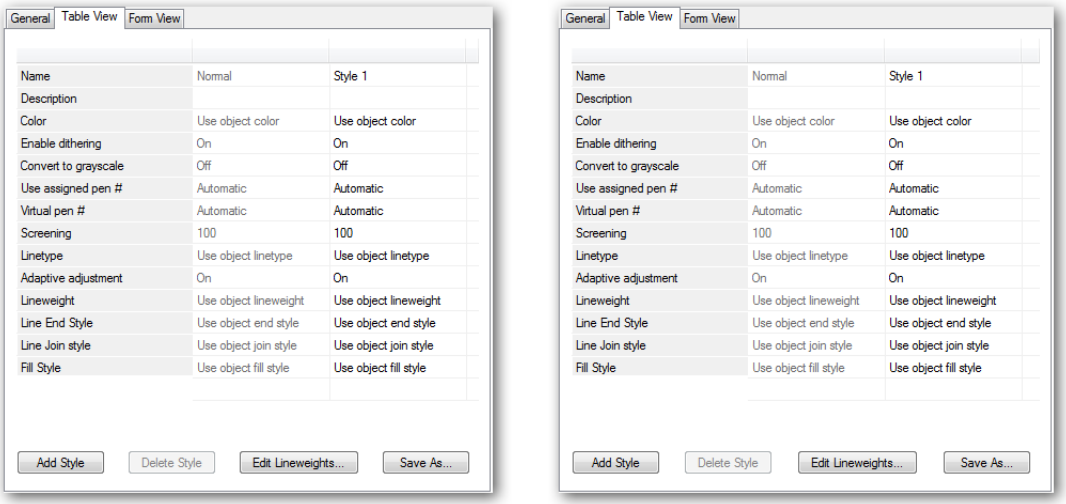
Left: *AutoCAD's multileader style editor;* ***right*** *BricsCAD's multileader style editor in Drawing Explorer*

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.



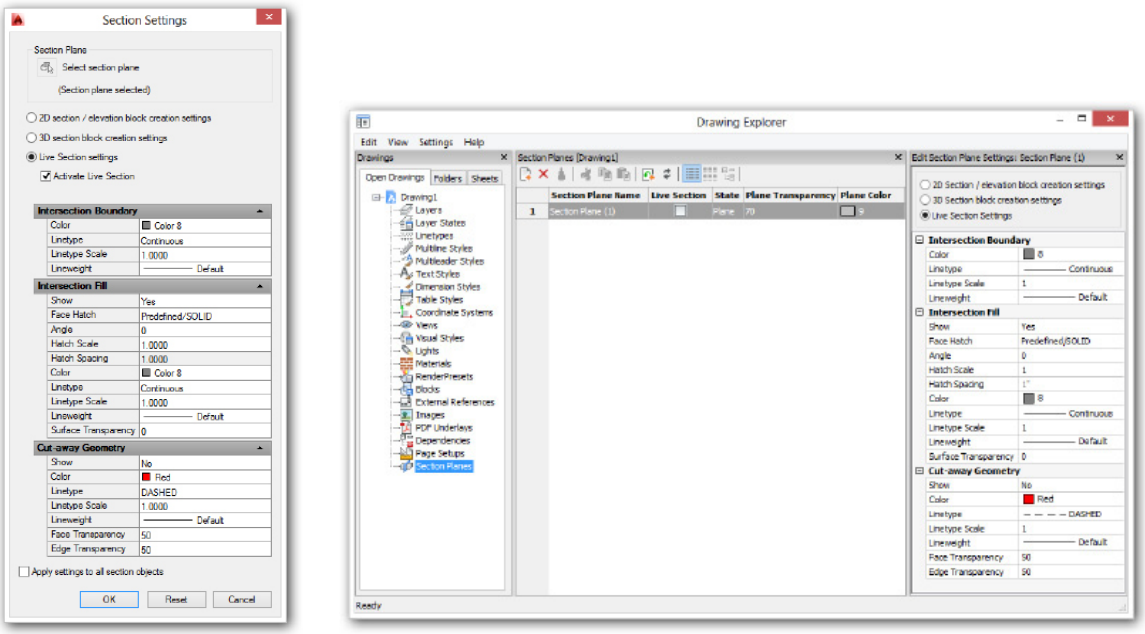
Left: *Plot style properties in AutoCAD... ;* ***right:*** *...and in BricsCAD.*

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

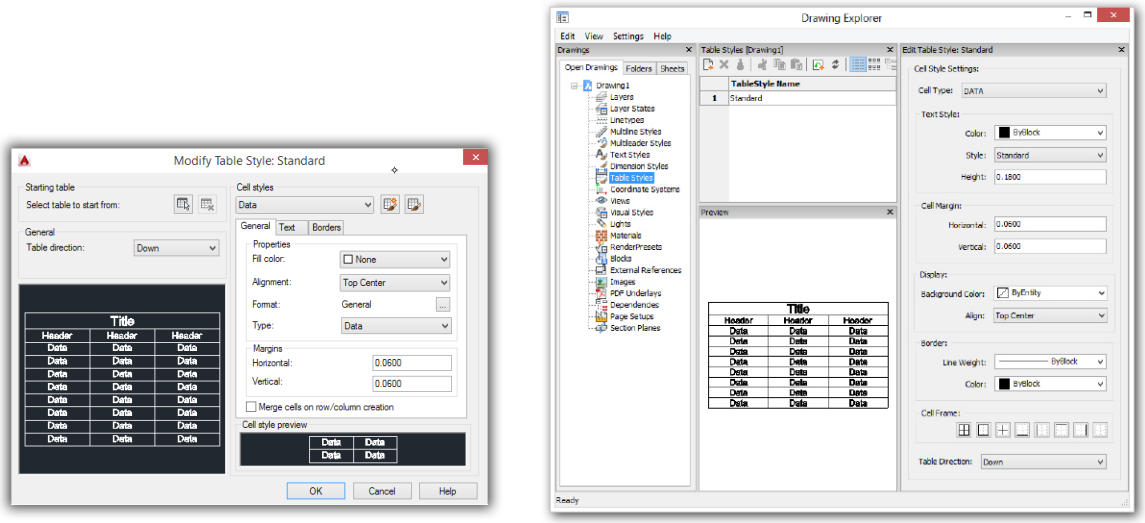
BricsCAD creates and edit table styles with the TableStyle command, as in AutoCAD. Unlike AutoCAD, BricsCAD's TableStyle command calls up the Table Style section of the ubiquitous Drawing Explorer.

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 in V16 has all the same named visual styles as AutoCAD, plus a few extras.

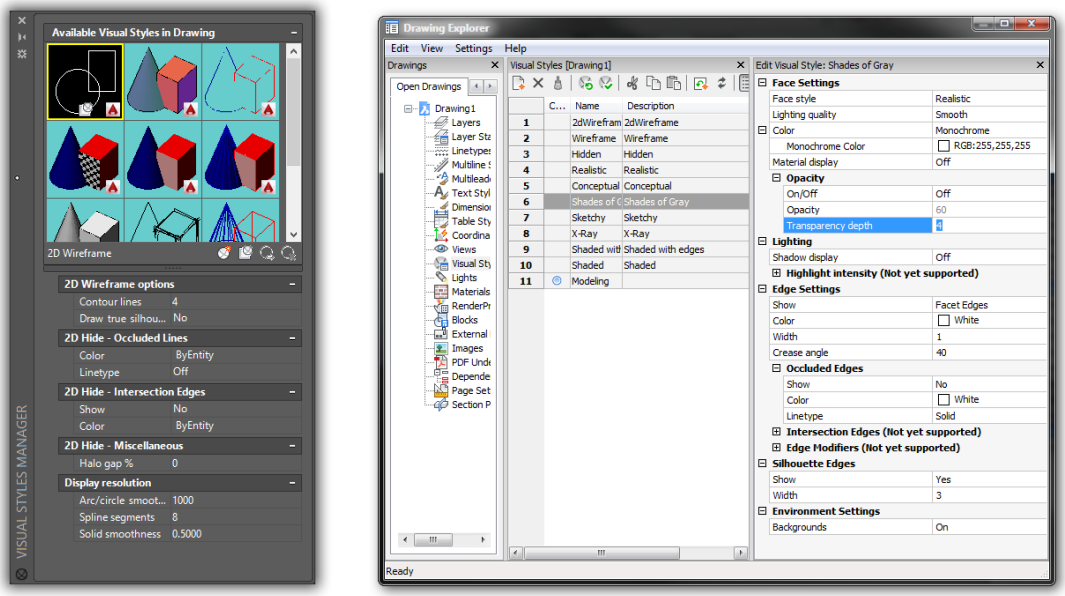
AutoCAD Command	BricsCAD Command
VsCurrent	ShadeMode
VisualStyles	VisualStyles
-VisualStyles	-VisualStyles

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
...	BIM
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 TO CUSTOMIZE BRICSCAD OR AUTOCAD, MOST OF THEIR ACTIVITY TAKES place inside a pair of dialog boxes accessed by these commands:

> **BricsCAD:** user customization through the **Settings** (alias: options) and **Customize** (alias: cui) commands
> **AutoCAD:** user customization 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 www.bricsys.com/estore/estoreBooks.jsp
- For detailed information on programming BricsCAD, refer to the online developer reference available free at www.bricsys.com/bricscad/help/en_US/V16/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 Butons**	**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 | Prifiles**	**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.

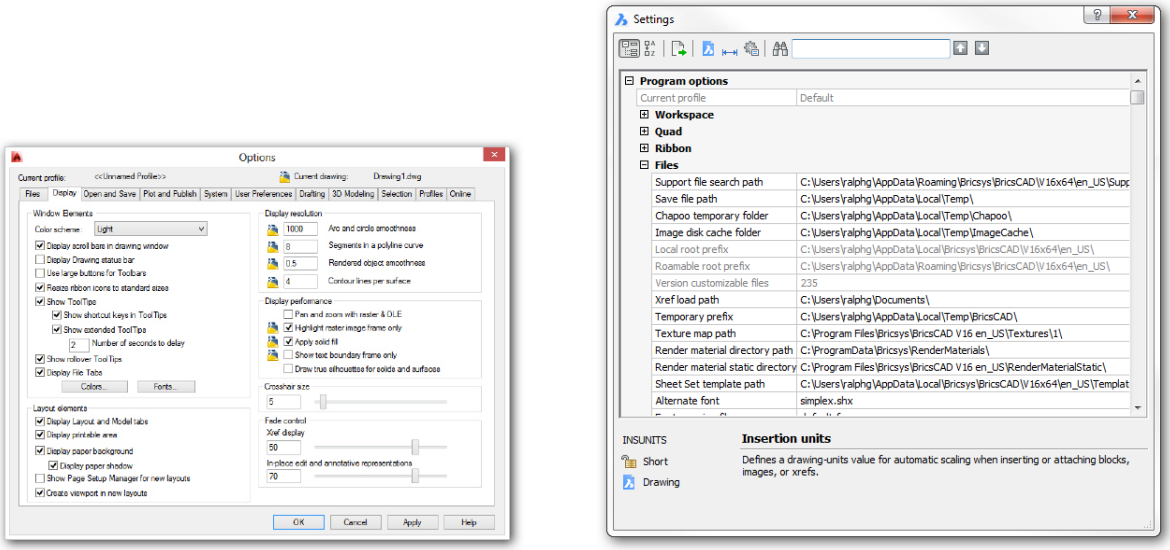
AutoCAD Options vs BricsCAD Settings

Just like AutoCAD, BricsCAD provides a set of extensive options for controlling your drafting environment — ever thing 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." (See Appendix B for the complete list of sysvars and preferences.)

Preferences operate just like system variables. Bricsys gave them the different name merely to indicate they are unique to BricsCAD.

Both CAD programs allow you to enter the names of sysvars and preferences directly at the command prompt. The old **SetVar** command is also available. 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, access the *acad.pgp* file with Notepad.)

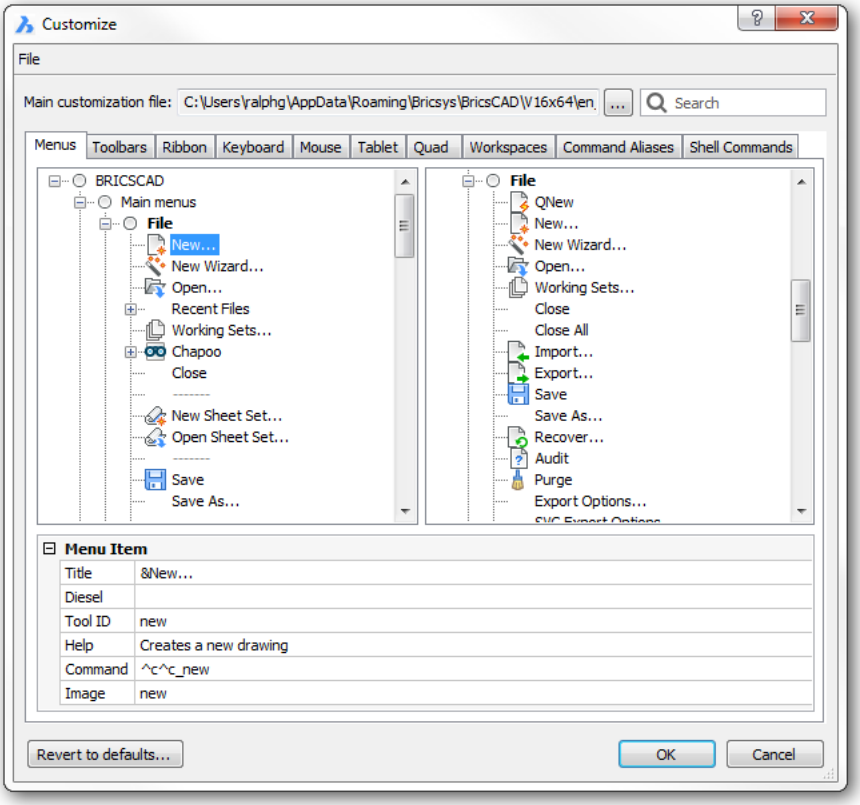
FILE PATHS

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 Commands

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 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

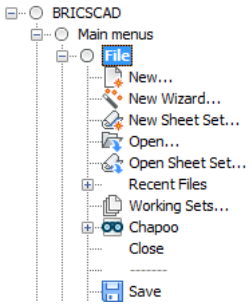
- **Keyboard** tab customizes the keyboard shortcuts
- **Mouse** tab customizes the mouse buttons, double-click actions
- **Tablet** tab customizes the tablet overlay menus and stylus buttons
- **Quad** tab customizes the quad cursor (not available in AutoCAD)
- **Workspaces** tab customizes the quad cursor context
- **Aliases** tab customizes the command aliases
- **Shell Commands** tab customizes the shell commands

The process for customizing each of these elements is nearly always identical. This means that when you learn the customization steps for one element (such as menus), you can then handle any other customization, such as context menus or toolbars. The way that BricsCAD customizes, however, is different from AutoCAD. So I show you an example of this, using BricsCAD's menus.

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**.

UNDERSTANDING BRICSCAD'S CUSTOMIZATION TREE

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.



Gray dots indicating container items

Here are more examples of containers:

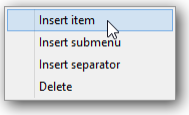
- **BRICSCAD** container holds the names of the *menu groups*. These groups include things like "Main Menus" (the menu items seen on the menu bar) and "Context Menus," which appear when you right-click entities.
  - **Main Menus** is a container that holds items that appear on the menu bar, such as "File"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

When you see a row of dashes that looks like "-----," this indicates the position of *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. Here is where you add, edit, and remove items to and from menus. To do so, go into the dialog box, right-click an existing menu item, and then choose an option from the shortcut menu that appears. Most actions are performed through choosing options from shortcut menus, like the one shown below.

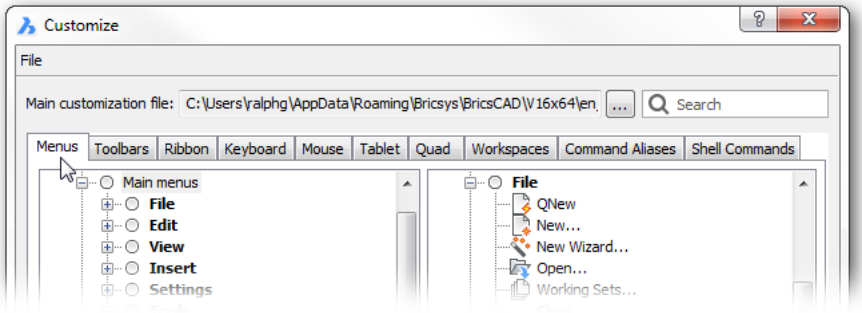


Right-clicking is how things get done in the Customize dialog box

Tutorial: Adding a Command to a Menu

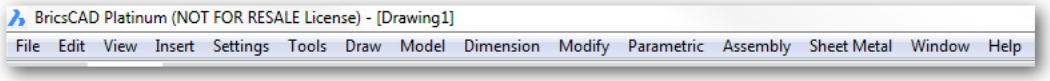
In this tutorial, you add the **CloseAll** command to the File menu. The CloseAll command closes all open drawings. It is to be located after the Close item. To change the menu, follow these steps:

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



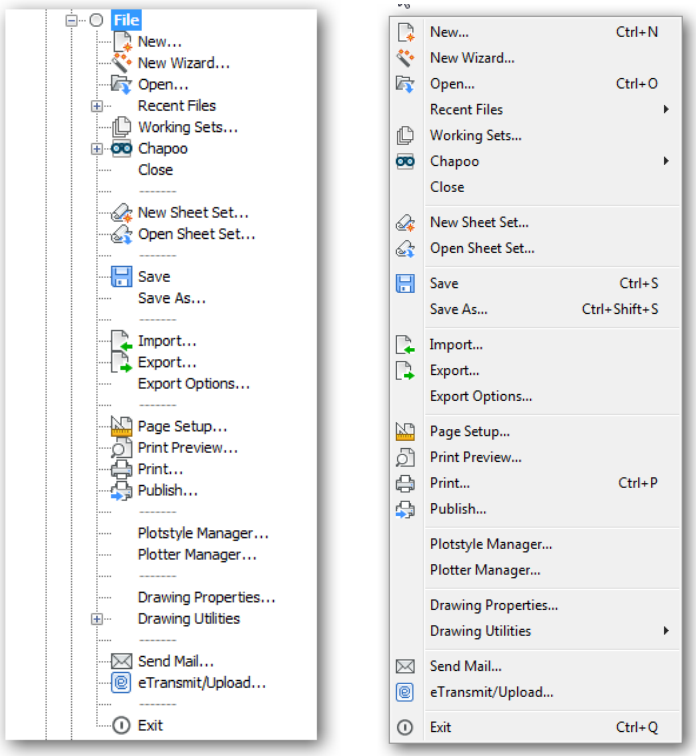
Accessing the Menu tab in the Customize dialog box

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.



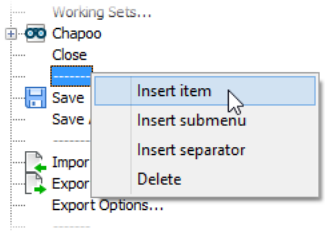
Names on the menu bar matching the list in the Customize dialog box, one for one

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.

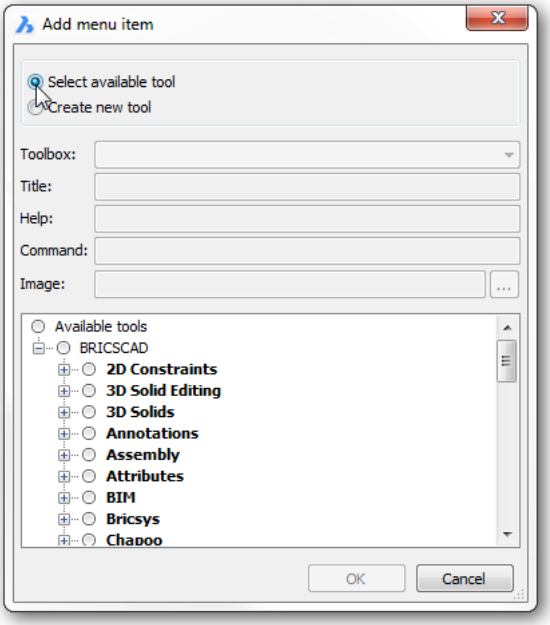


Inserting an item above the selected one

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**.

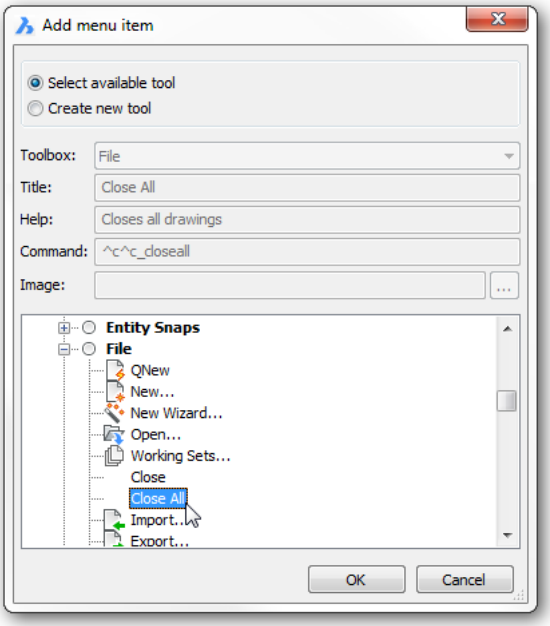
   a. Choose the **Select Available Tool** option to access all of BricsCAD's built-in commands. (The other option, **Create New Tool**, is for creating macros — two or more commands strung together.)



Dialog box for creating new menu items

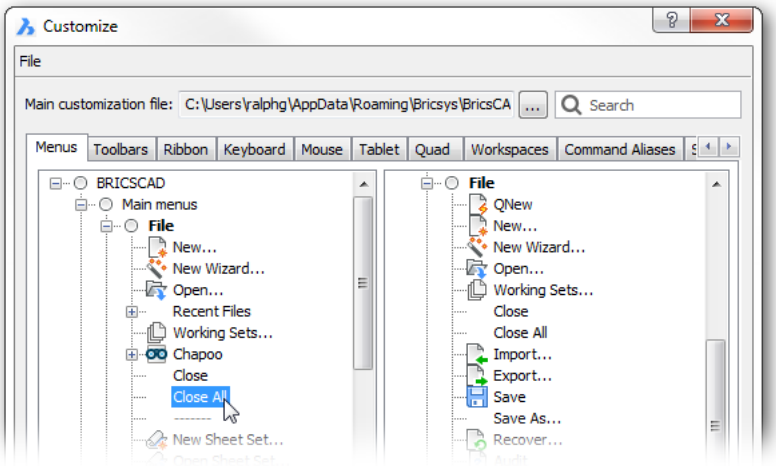
   b. Under Available Tools, scroll down to the **File** item. The fast way to get there is to click any item in the list (such as "2D Constraints"), and then tap the **F** key on the keyboard.

   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.



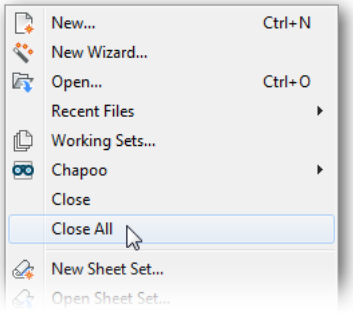
Choosing "Close All" from the list of a available commands

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 the 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.

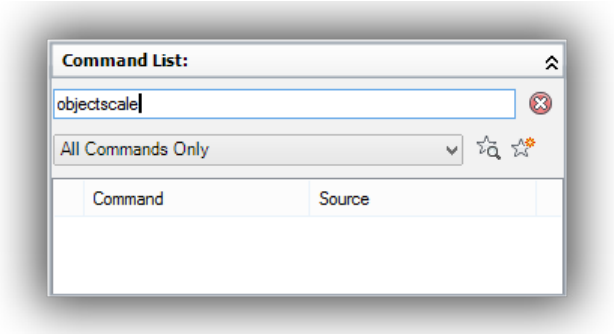
TIPS FOR WORKING WITH BRICSCAD'S MENUS

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

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

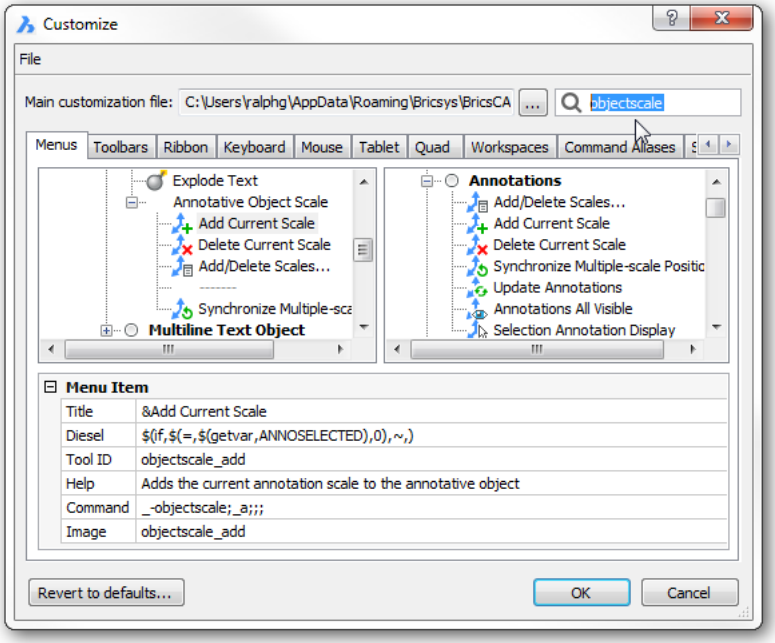
All commands. To see all commands in BricsCAD, peruse the list found the Customize dialog box.

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"?

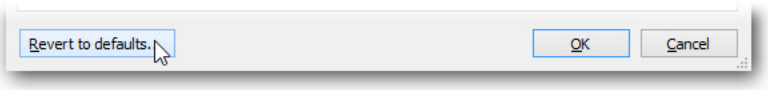
Sometimes a shortcut menu shows the verb **Append**, other times it shows **Insert** instead. The words seem similar, but have different actions; here's how:

- **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.

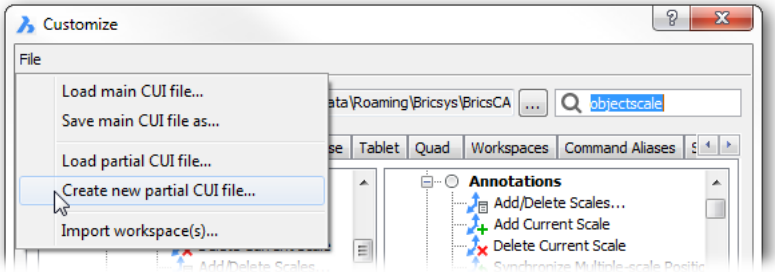
TIP The correct way is to add commands is to first create a new partial menu for them. The reason you do this is because of the **Revert to Defaults** button, which you see at the bottom of the Customize dialog box. Should a user (or you, even) click this button, then all customizations are lost! Except,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

In BricsCAD, this is how you construct macros in a partial menu.

Step 1: How to Create Partial Menus in BricsCAD

First, create the new partial menu.

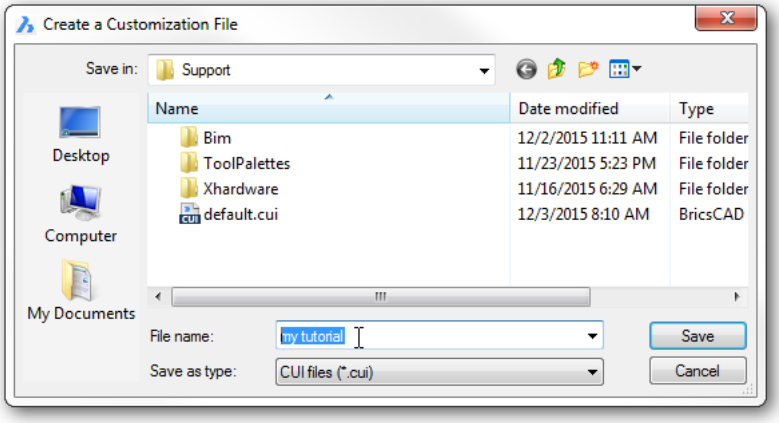
1. Click **File** in the Customize dialog box, and then choose **Create New Partial Cui File.**



Creating a new partial CUI file

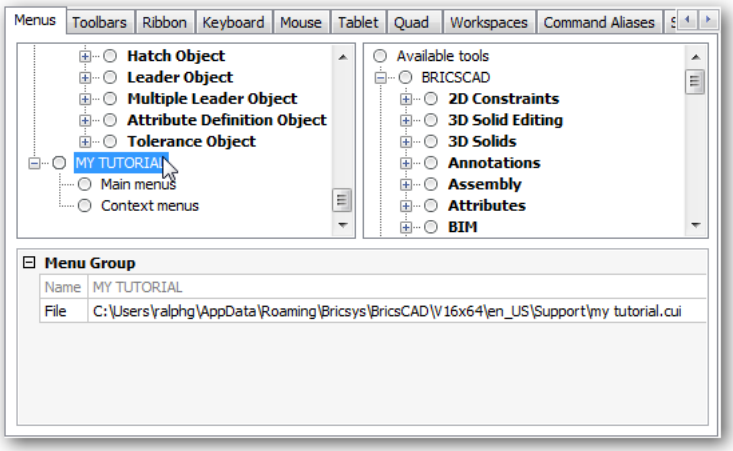
2. Notice the Create a Customization File dialog box. In the **File Name** field, enter a name that is brief but descriptive. For this tutorial, enter "my tutorial."

```
File Name       my tutorial
```



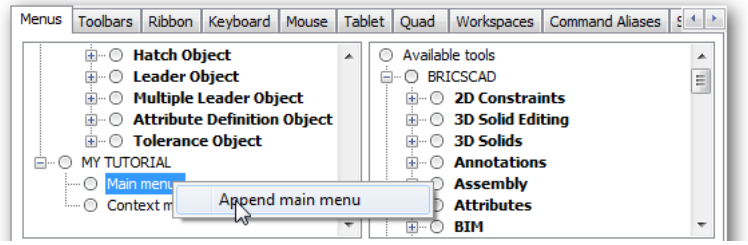
Naming the new partial menu file

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.)



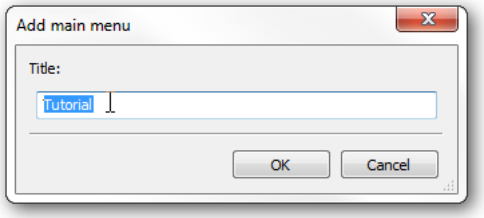
My Tutorial added to menus

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



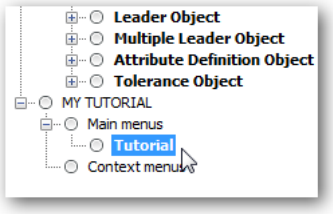
Adding a new main menu item

5. In the Add Main Menu dialog box, give the new menu a name, like "Tutorial," and then click **OK**.



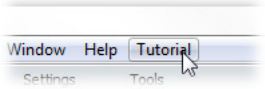
Naming the new menu item

Notice the new Tutorial menu item.



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

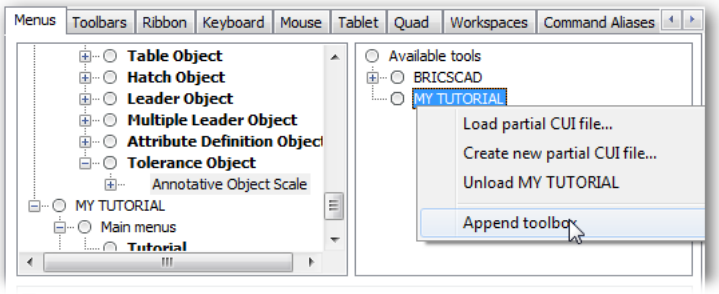
The partial menu is ready for the next step: adding custom commands.

Step 2: Adding Custom Commands to BricsCAD

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

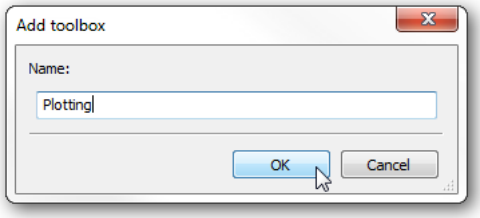
1. Go to the right-hand side of the Customize dialog box, and in the **Available Tools** pane, navigate to the "My Tutorial" partial menu.

2. New commands are collected into *toolboxes* in partial menus. A toolbox is a collection of similar commands, such as ones related to drawing or to file management. Right-click "My Tutorial," and then from the shortcut menu, choose **Append Toolbox**.



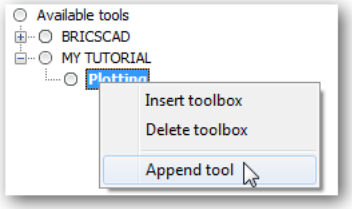
Adding a new toolbox to a new partial menu

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



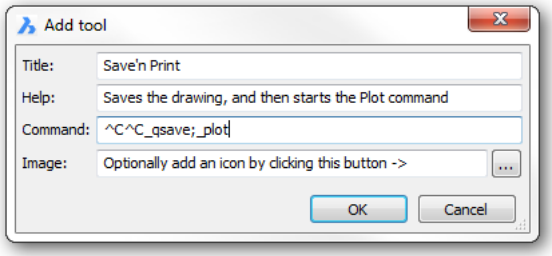
Naming the new toolbox

4. With the toolbox created, go ahead and create the new tool: right-click and choose **Append Tool**.



Adding a tool to the toolbox

4. In the Add Tool dialog box, define the new command or macro in the **Command** field.



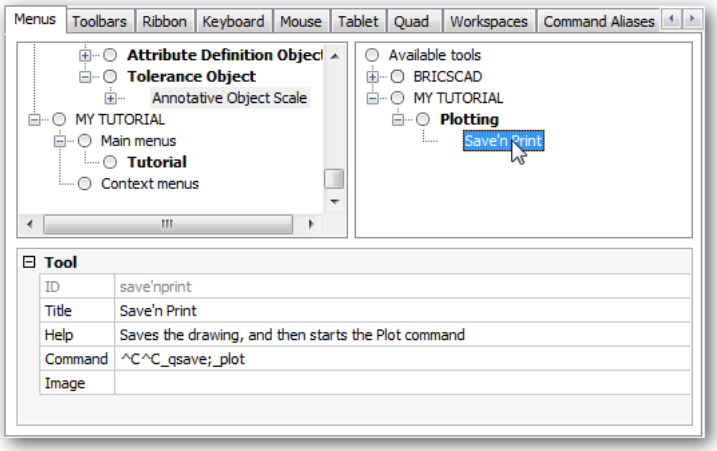
Fields for defining parameters of new tool

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 Plot command.	Specifies the help text that appears on the status bar
Command	^C^C_qsave;_plot	Specifies the macro that cancels the current command, saves the drawing, and then starts the Plot command
Image	*(leave blank)*	Specifies the icon, although none is required for menus

5. 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 with regular commands.

6. Click **OK** to exit the Customize dialog box.
7. 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:

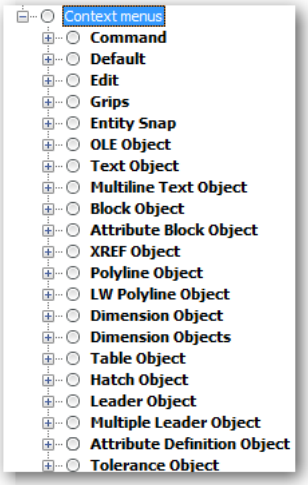
& (*ampersand*) designates shortcut keystrokes for accessing menu items with the **Alt** key

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

About Diesel and DCL in BricsCAD

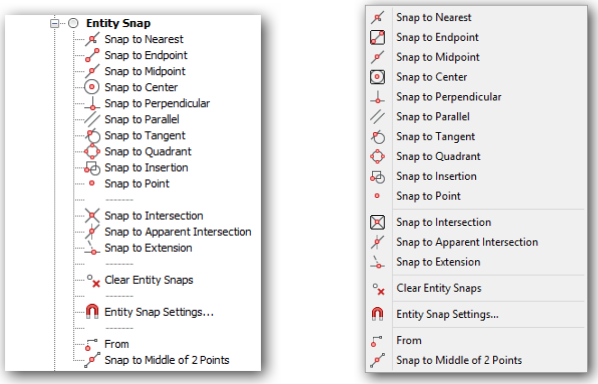
AutoCAD and BricsCAD employ the same Diesel expressions in menu macros and LISP routines, and the same DCL (dialog control language) code for constructing dialog boxes.

CUSTOMIZING CONTEXT MENUS



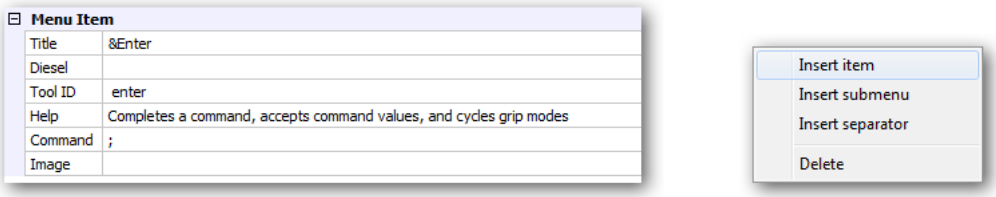
BricsCAD calls shortcut menus "context menus," because the menus change their content depending on the context. Context menus are found in the **Menus** tab, below the **Main Menus** section.

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. Here is what it look like:



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

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



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

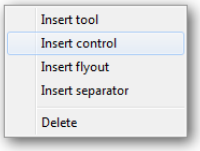
CUSTOMIZING TOOLBARS

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

(The visibility of AutoCAD's toolbars is controlled by the current workspace; all its toolbars are by default turned off. In contrast, a selection of toolbars are by default turned on in BricsCAD.)

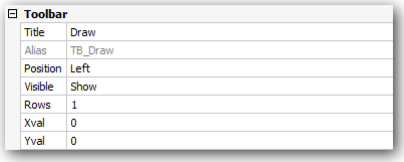
The process for customizing toolbars is identical to that of customizing menus, with two exceptions:

- 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:

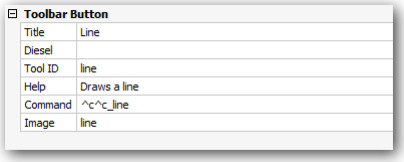


Parameters for positioning toolbars

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:

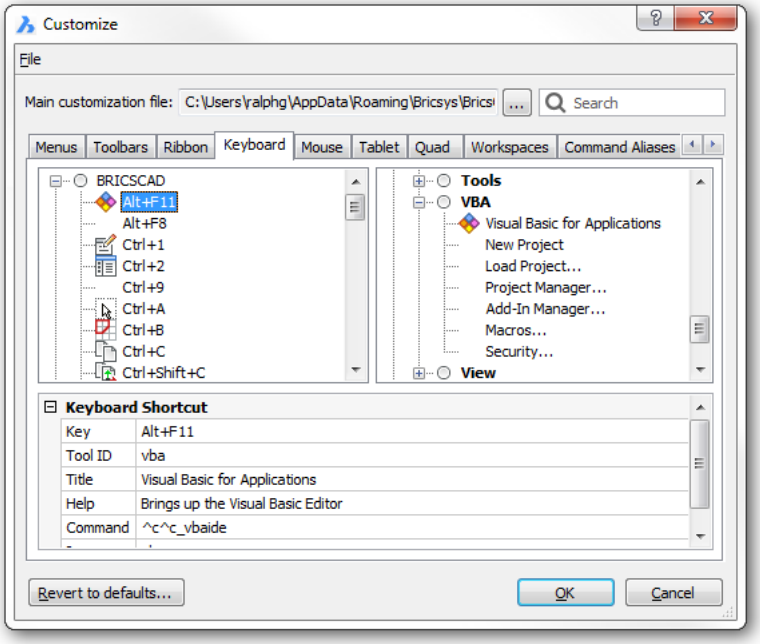


Parameters for toolbar buttons

Similar properties are available for flyouts.

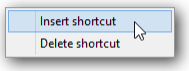
CUSTOMIZING KEYBOARD SHORTCUTS

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.



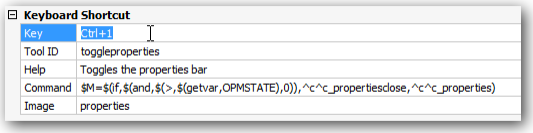
Customizing shortcut keystrokes

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:

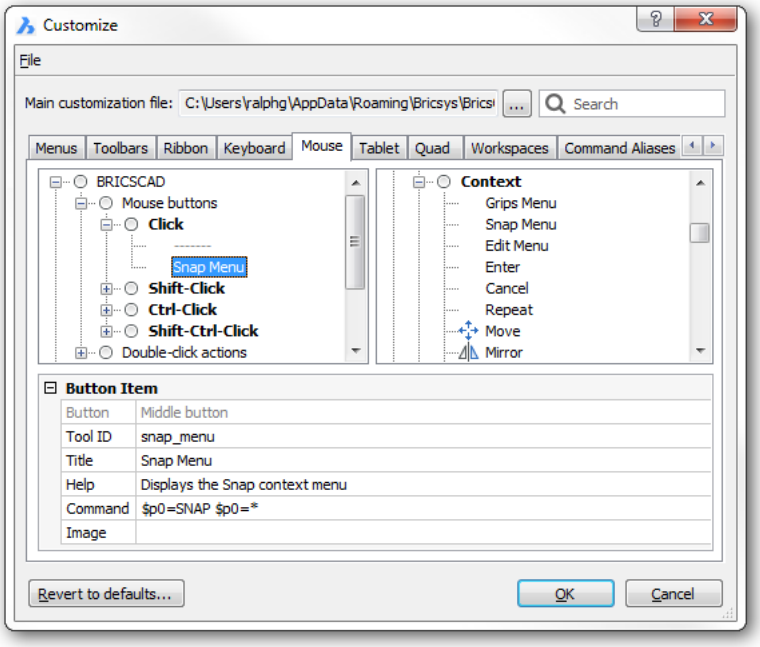


- **CTRL** keys
- **SHIFT**+**CTRL** keys
- Function keys
- **SHIFT**, **CTRL**, **ALT**, **CTRL**+**ALT**, **SHIFT**+**ALT**, **SHIFT**+**ALT**, and **SHIFT**+**ALT**+**CTRL** function keys

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

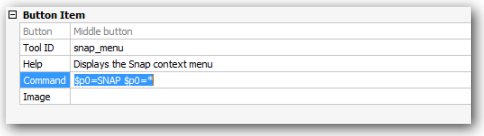
CUSTOMIZING MOUSE BUTTONS & DOUBLE-CLICKS

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

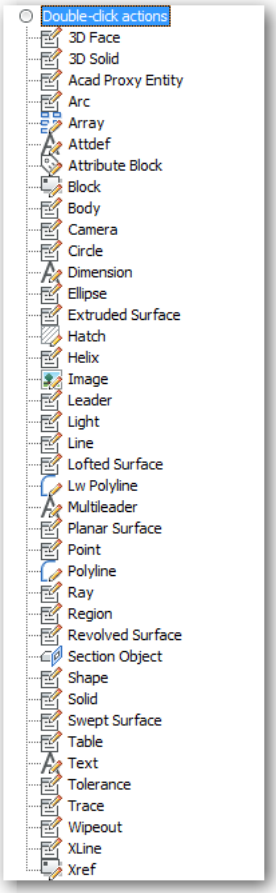


Customizing mouse button actions

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:

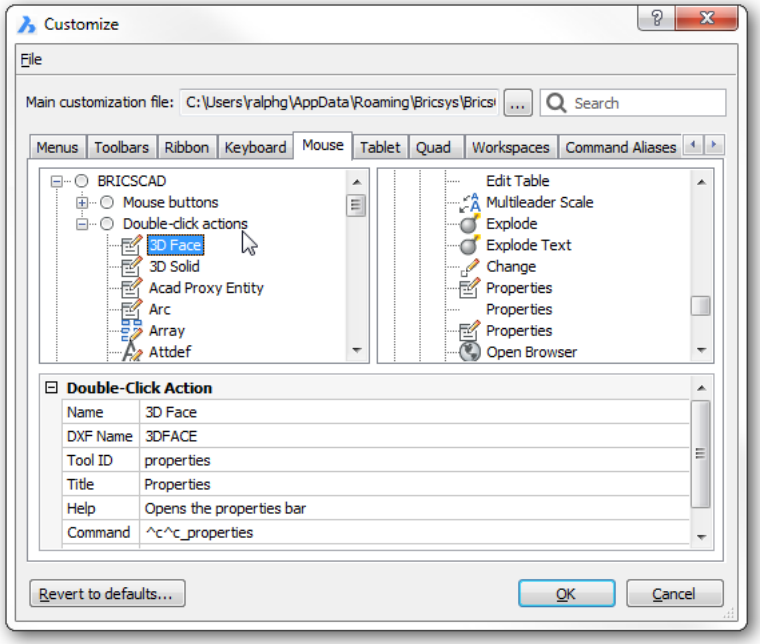


Editing actions for mouse buttons



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.



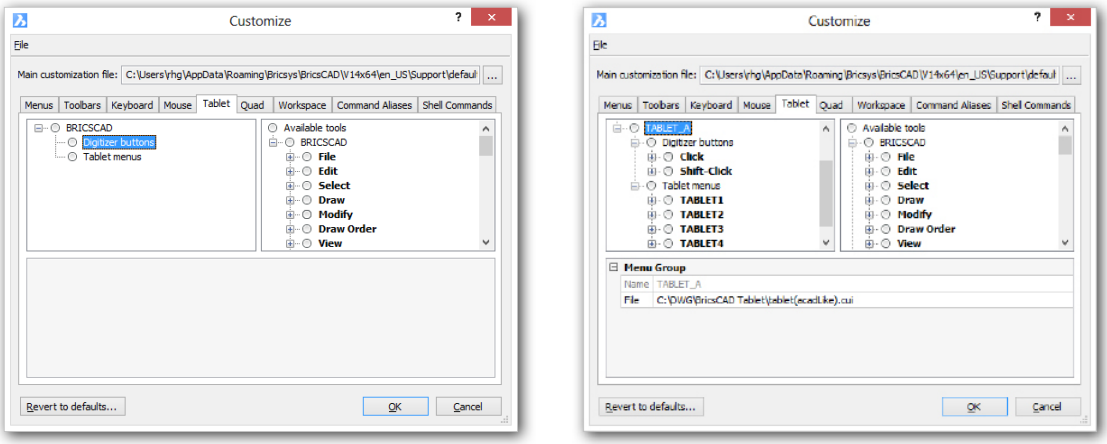
Customizing double-click actions in BricsCAD

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.

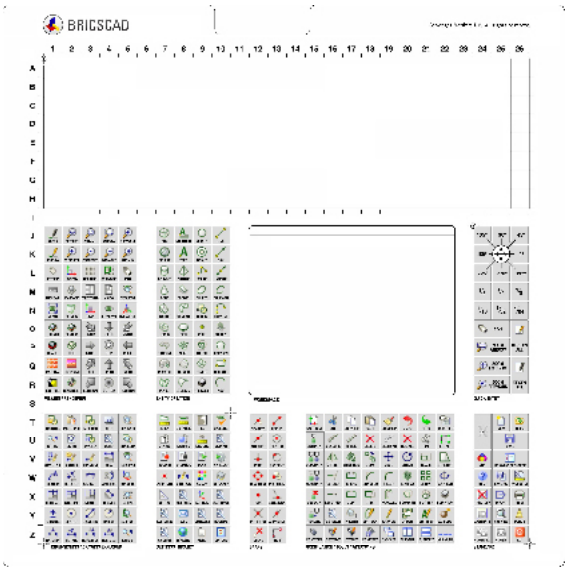
CUSTOMIZING TABLET BUTTONS AND MENUS

Tablet overlay menus and digitizer buttons are customized in BricsCAD through the **Tablet** tab, as illustrated below. Entries under **Digitizer Buttons** and **Tablet Menus** look empty because no tablet menu is loaded with the Default profile, as shown to the left, below.



Left: *Default items for digitizer buttons and tablet overlays in BricsCAD*
Right: *Items after a partial CUI file for tablets is loaded in BricsCAD*

To add tablet support to BricsCAD, download CUI files and drawings for tablet buttons and overlays from www.bricsys.com/bricscad/tools/Tablet.zip . The tablet overlay drawing provided by Bricsys is illustrated below:



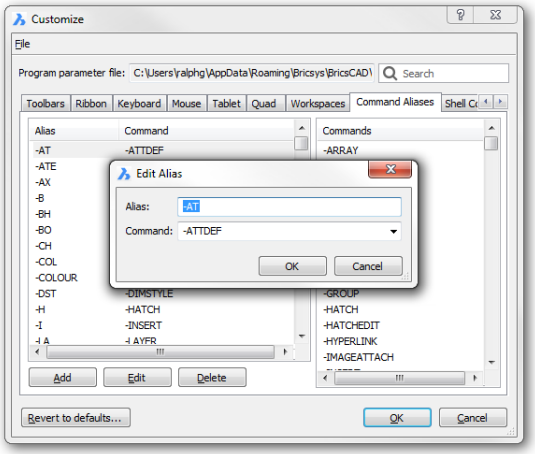
Tablet drawing provided by Bricsys containing the same commands as AutoCAD

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 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:

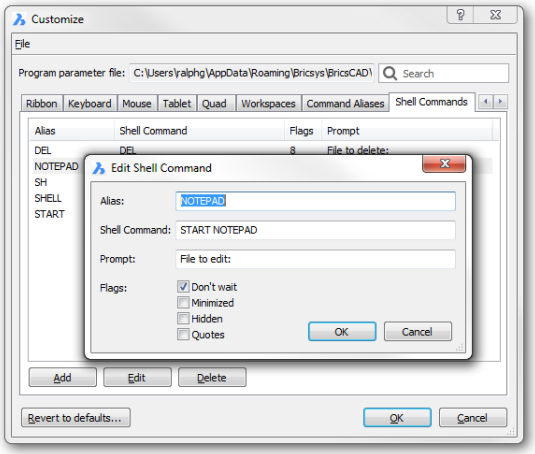
BricsCAD aliases are stored in the *default.pgp* file.

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

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

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

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.

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*

SHX Fonts

AutoCAD keeps SHX fonts in the *C:\program files\autodesk\autocad 2016\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 V16\Fonts*
- Linux stores SHX fonts in folder */opt/bricsys/bricscad/fonts*
- Mac stores SHX fonts in folder */Applications/BricsCAD V16.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 2016\R20.0\enu\Support
BricsCAD Windows	*default.fmp*	C:\Users\login\AppData\Roaming\Bricsys\BricsCAD\V16\en_US\Support
BricsCAD Mac	*default.fmp*	Users\login\Library\Preferences\Bricsys\BricsCAD\V16x64\en_US\Suppot
BricsCAD Linux	*default.fmp*	home/login/Bricsys/BricsCAD/BricsCAD/V16/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:

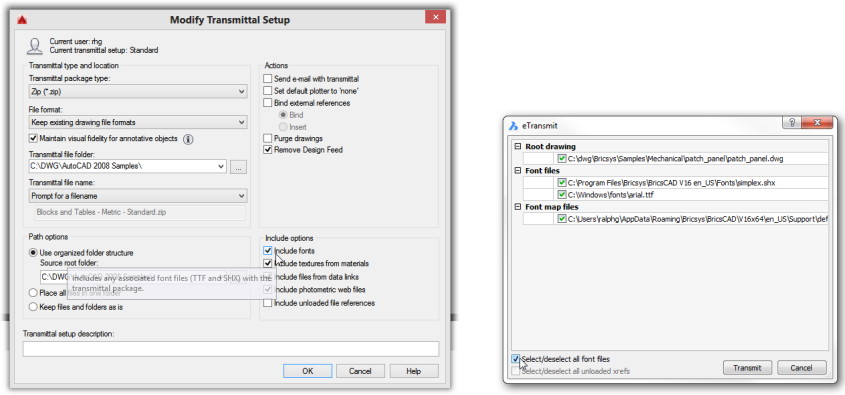
```
ic-comp;complex.shx
ic-complex;complex.shx
ic-gdt;gdt.shx
ic-ital;italic.shx
ic-italc;italicc.shx
```

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 2016\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\V16\en_US\Support*
- Linux stores the files in folder *home/<login>/Bricsys/BricsCAD/BricsCAD/V16/en_US/Support*
- Mac stores the files in folder */Users/<login>/Library/Preferences/Bricsys/BricsCAD/V16x64/en_US/Support*

Tutorial: How to Copy AutoCAD .**lin** and .**pat** Files to BricsCAD

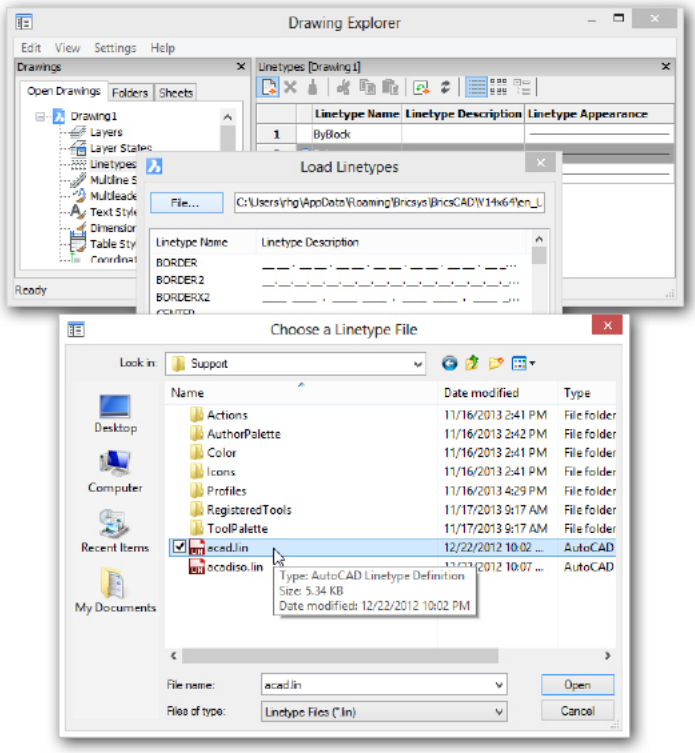
If you wish to reuse linetypes and hatch patterns from AutoCAD, then follow these steps to copy and rename them:

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 2016\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

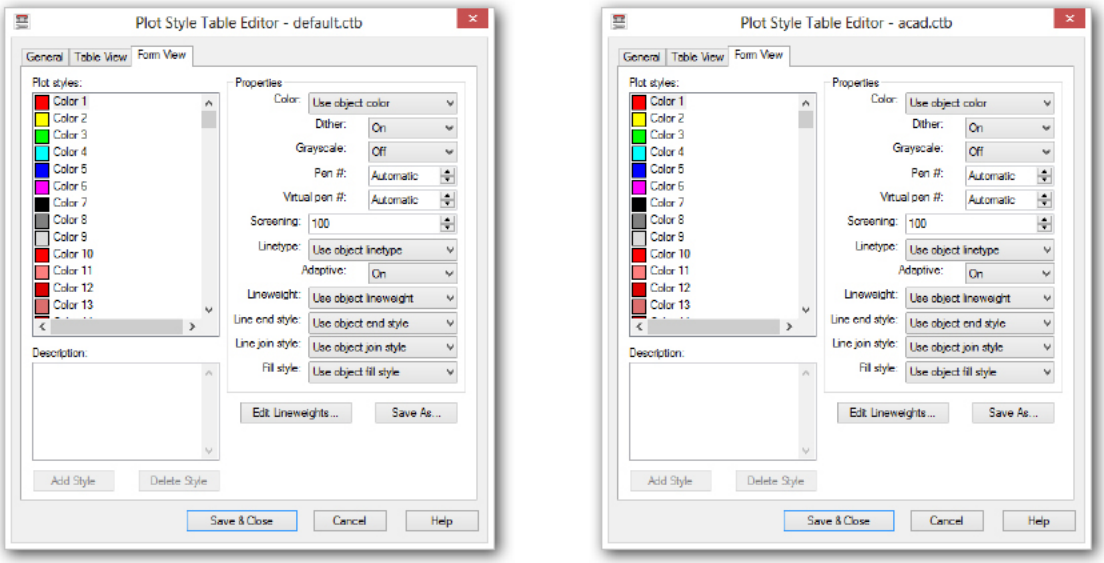
BricsCAD and AutoCAD support both color-based and style-based plot styles that allow entities to look different when plotted. Recall that CTB files are for the older color-based plot style tables, while STB files are for the newer style-based plot style tables. The figures below show that the style-based 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

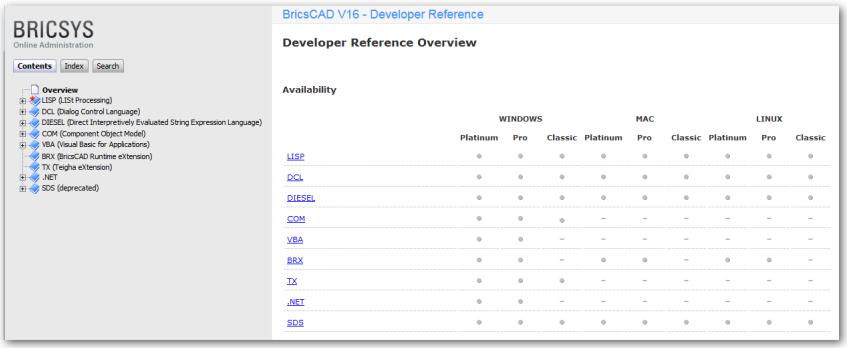
Programming Considerations

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/V16/DevRef.



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 V16 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/V16/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 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. Releases of BricsCAD prior to V8 use VBI files, which can be converted to VBA for V8 and later. In BricsCAD, VBA works only on the 32-bit version. VBA is not available in BricsCAD for Linux and Mac, or the 64-bit version of Windows.

See the online VBA conversion tool that Bricsys provides at http://www.bricsys.com/common/vbaconversion.

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.

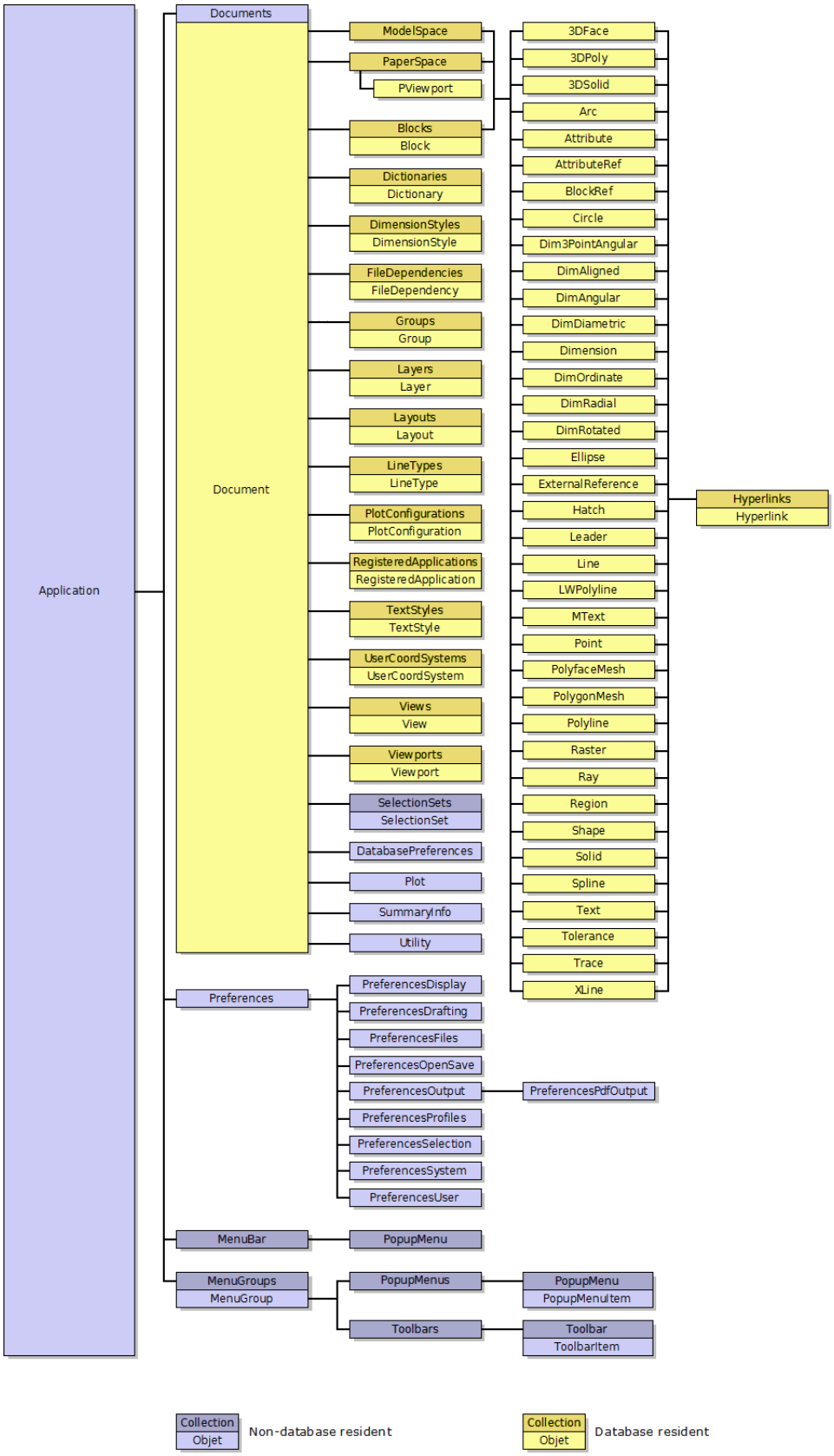
(ADS is short for AutoCAD Development System, the first API for AutoCAD to use external libraries. SDS is short for SoftDesk Development System, a workalike first developed by SoftDesk for its IntelliCADD project.)

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/V16/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++.

BricsCAD V16 Automation Object Model

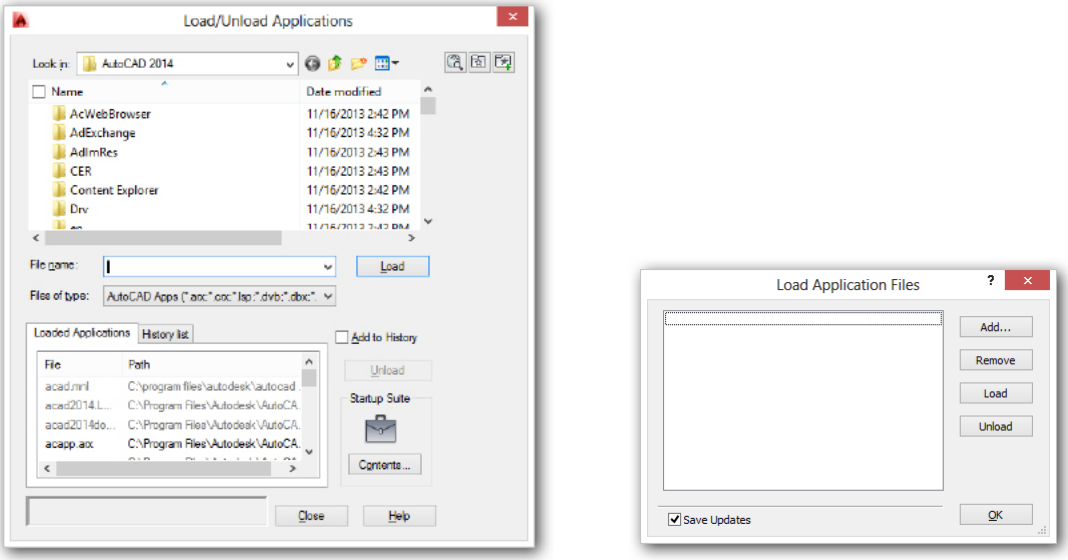


Though BricsCAD's object model is quite similar to AutoCAD's, it is not identical. Nevertheless, most VBx code written for AutoCAD should work directly under BricsCAD. When you find a required element missing from the object model, the BricsCAD developer support team is open to creating the functions 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.



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 BRICSCAD AND other CAD systems, such as AutoCAD. This chapter explores the realities of running a dual-CAD shop, and explains how to solve issues.

We examine the benefits and drawbacks to running the Linux operating system as a cost saving alternative to Windows or OS X, and touch on the place of CAD on mobile devices and in Web browsers.

Reasons for Using Multiple CAD Systems

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, paying $4,195 for every legal copy of AutoCAD was too expensive, and so they run the 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 package, 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 2016	$419,500	
Mix of AutoCAD and BricsCAD		$319,050
10 of AutoCAD 2016	$ 41,950	
90 of BricsCAD Pro V16	$ 58,500	
Pure BricsCAD		$354,500
100 of BricsCAD Pro V16	$ 65,000	

The actual cost to license one hundred seats will be lower than shown by the table, because both CAD vendors offer better prices on bulk purchases and network licenses.

I chose to list the price of BricsCAD Pro rather than Platinum, because the additional functions provided in Platinum are not found in AutoCAD, such as 3D constraints and assemblies.

Upgrades. Following the initial licensing cost, design firms can choose to spend on other charges typically associated with software use:

- Upgrade fees
- Annual maintenance or support fees (include upgrades at no added cost)

I don't compare upgrade fees or maintenance, because Autodesk had been charging the full list price ($4,195) for upgrades, and as of January 31, 2016 eliminated AutoCAD upgrades and perpetual licenses all together. In this category, BricsCAD also has the advantage.

Subscriptions. The third alternative is to purchase subscriptions instead of perpetual licences. The advantage is that design firms don't face the large, up front financial burden; the disadvantage is that subscriptions become the more expensive after just 3.5 years (AutoCAD) or 2.7 years (BricsCAD).

This table shows the cost for subscribing to 100 licenses on one-year plans, the only length common to both CAD programs:

Number of Seats	Annual Subscription Price	Annual Cost Savings
Pure AutoCAD		$ 0
100 of AutoCAD	$118,776	
Mix of AutoCAD and BricsCAD		$ 85,749
10 of AutoCAD	$ 11,878	
90 of BricsCAD	$ 21,150	
Pure BricsCAD		$ 95,276
100 of BricsCAD	$ 23,500	

No mixing allowed! To be on subscription, Bricsys requires all seats at a single site be on subscription.

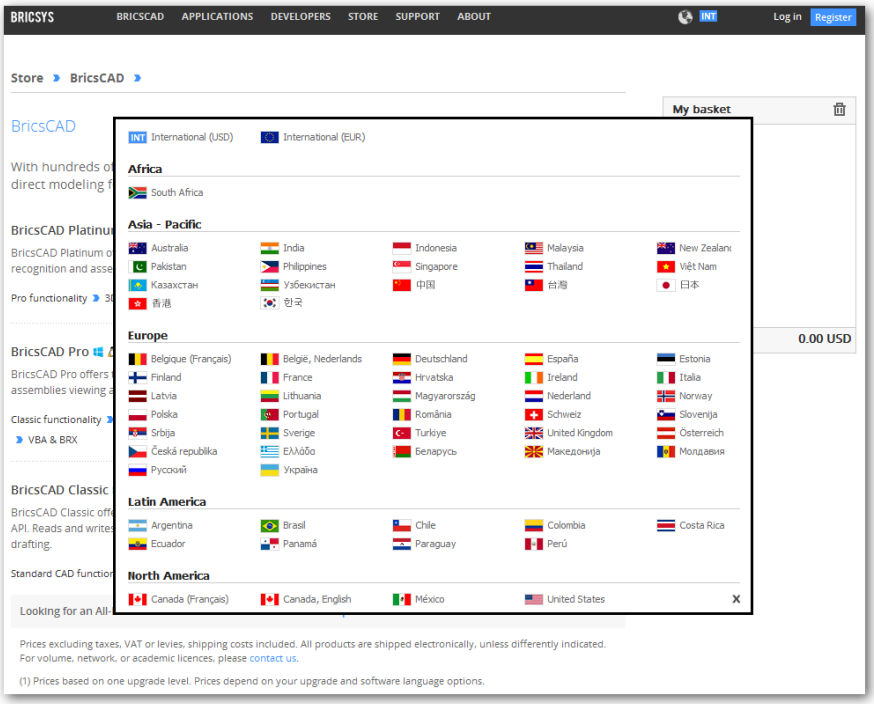
Hardware. I did not include the benefit of using older and slower hardware with BricsCAD, as this cannot be easily quantified financially; there are too many variations in workstations and pricing. BricsCAD does not need the more expensive computers and graphics boards that AutoCAD requires to run well.

Nevertheless, the advantage goes to BricsCAD, as initial hardware costs are lower and subsequent hardware upgrades are rarer.

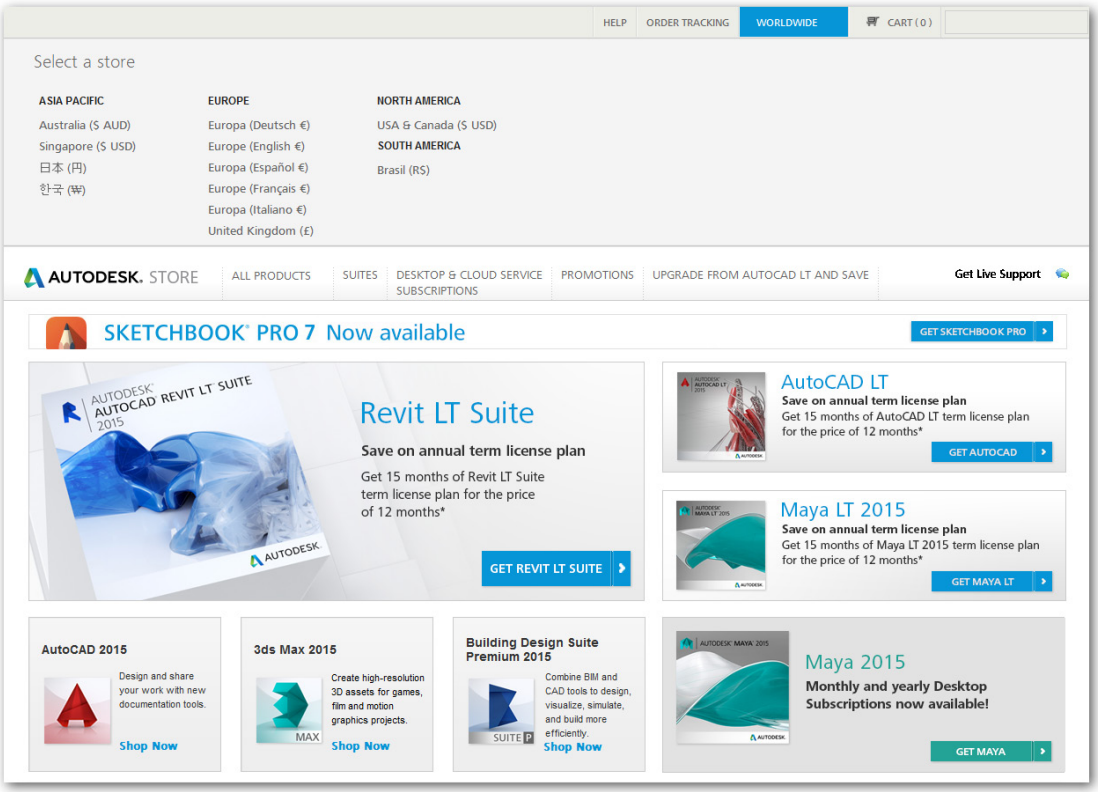
Country-Biased Pricing

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: www.autodesk.com/store
- Bricsys: www.bricsys.com/estore/estoreBcad.jsp



The online store from Bricsys



The online store from Autodesk

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 major 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 then 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.

Solutions to High License Fees

For these firms, the solution is to license mostly lower-cost products, specifically AutoCAD LT, BricsCAD, and the like. Indeed, BricsCAD 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. (OS X from Apple is free, but runs only on Macintosh-branded computers, which tend to be the most expensive kind.) 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.

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.*

By copying files in whole, 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.

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 effectivbly 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 will produce the highest profits with the fewest expenses — measurable and unmeasurable. Unmeasurable expenses include the difficulty of using a software system. In the case of CAD, this means 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 its 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 some years, Autodesk put huge resources into leap-frogging AutoCAD ahead of the competition, making the "100% Pure DWG" situation more true. In recent years, however, Autodesk has slowed its pace, with AutoCAD 2014 onwards gaining only a few new functions each year, and the file format remaining unchanged for more than three years.

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 not unlike a firm 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's counterweight to Autodesk is the Open Design Alliance. The ODA was established in the late 1990s to document Autodesk's DWG format, which it kept proprietary by not documenting its content. Today, the organization has 1,200 members. It provides APIs to member organizations that allow their software to read and write AutoCAD DWG, MircoStation DGN, and Adobe PDF files, as well as other aspects of CAD, such as an equivalent to the ARX programming interface, ADT and MDT object enablers, and the licensing of add-on software, such as the ACIS solid modeling kernel. 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 ODA, Autodesk broke down and documented DWG by licensing 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 keep the workalikes from replicating all these AutoCAD's functions in their entirety; on their own, they don't have the programming resources. Banded together under ODA, however, they can make good progress.

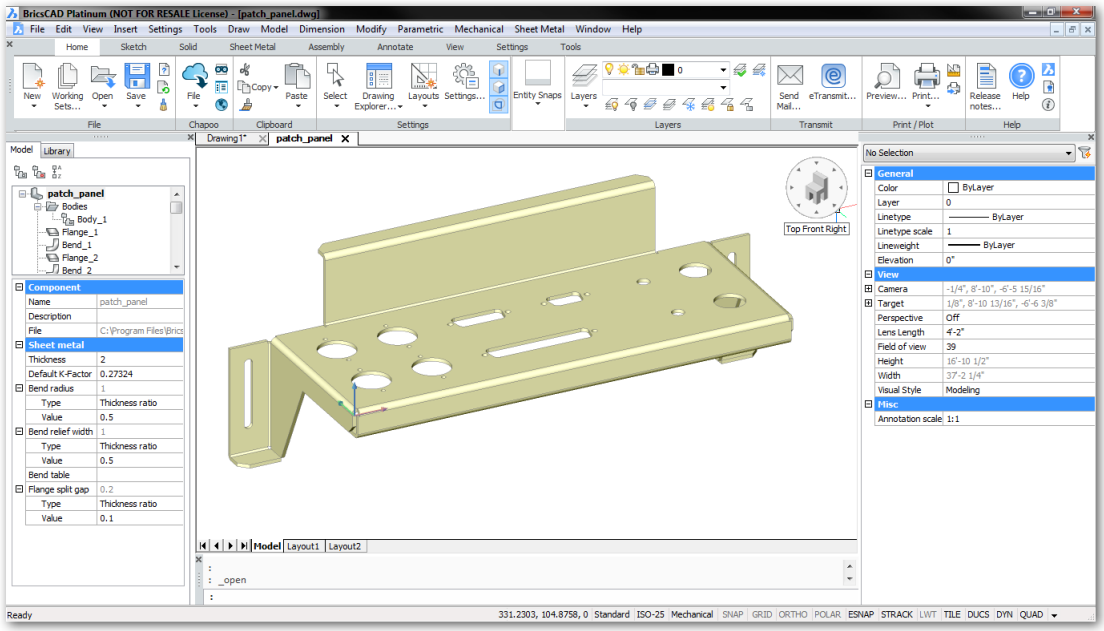
The ultimate good news for workalikes is that there is no need to replicate AutoCAD completely. It turns out that 3D point clouds and such are of little interest to heads-down drafters. If a design firm needs the capability, there are many third-party stand-alone products that do as good a job as AutoCAD — or better. The majority of AutoCAD and BricsCAD users produce 2D drawings.

A Solidworks product manager proclaimed at a recent conference, "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 this area of 3D. 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.



Bricsys doing sheet metal design

BricsCAD is on its way to becoming something like an AutoCAD-compatible version of Inventor, but for under $2,500. Indeed, the Bricsys CEO aims to be dominant in the MCAD market by 2020.

Running BricsCAD & AutoCAD in One Office

To run more than one CAD system in your office successfully, it is crucial that you first take the time to 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 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 detail 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 back in 2011; I've updated it to reflect added capabilities in more recent releases of BricsCAD. 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.~~

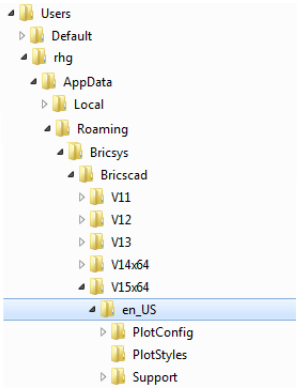
Common files are stored on the computer you use; these are files, such as font files and printer drivers, that are common to many programs. CAD programs make use of these files.

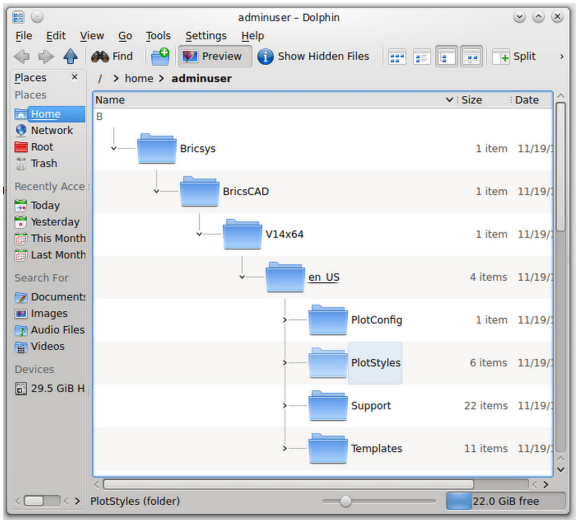
Temporary files are stored "anywhere," locally or on the network; these files are created by CAD programs for the duration of the editing session, such as automatic backup files.

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/V16
```

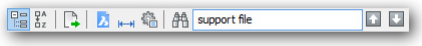
TIP BricsCAD V16 adds the following commands to make it easier to handle support files:
SupportFolder opens the *C:\Users\<login>\AppData\Roaming\Bricsys\BricsCAD\V16x64\en_US\Support* folder.
TemplateFolder opens the *C:\Users\<login>\AppData\Local\Bricsys\BricsCAD\V16x64\en_US\Templates* folder.
WhoHas 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.

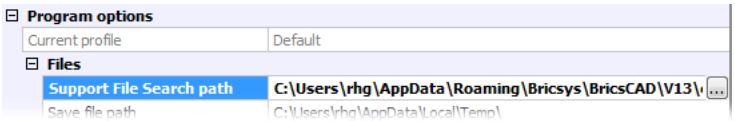
To direct BricsCAD to use AutoCAD's support files, open the Settings dialog box, and then access the **Program Options**, like this:

1. Start BricsCAD, and then enter the **Settings** command.
2. In the Search field, enter **support file**.



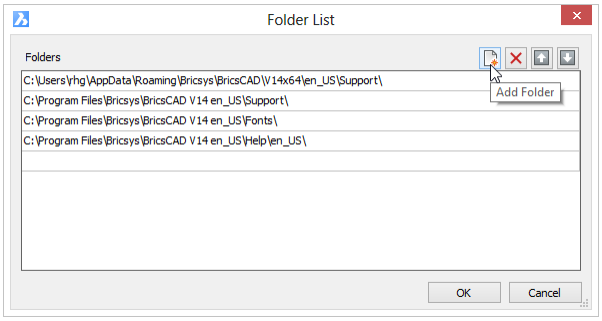
Searching for the phrase "support file"

3. Notice that the Settings dialog box jumps to the Support File Search Path item.



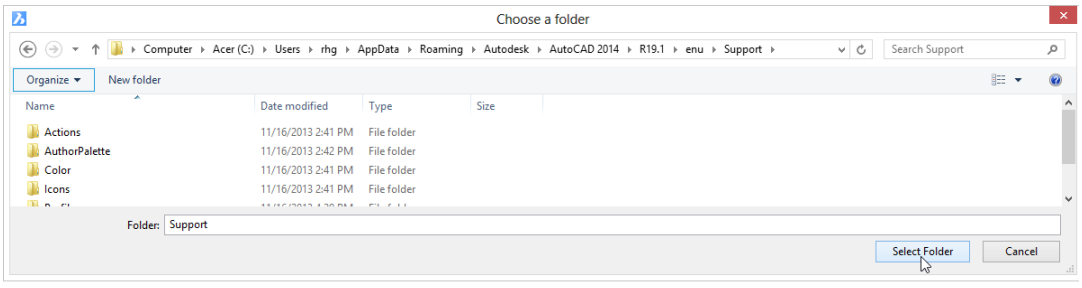
The support file search path entry in the Settings dialog box

4. Click the … **Browse** button. Notice the Folders List dialog box.



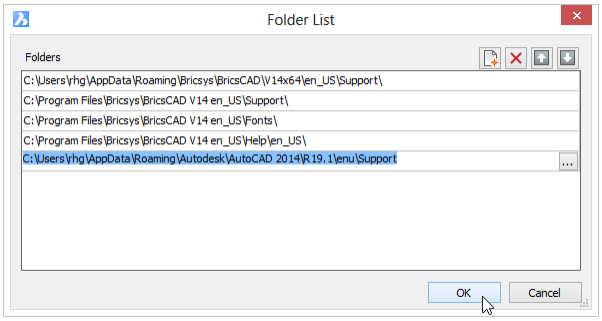
Adding folders to BricsCAD's search path

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 2016 - English\R20.0\enu\Template*
   Most other support files at *C:\Users\<login>\AppData\Roaming\Autodesk\AutoCAD 2016 - 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

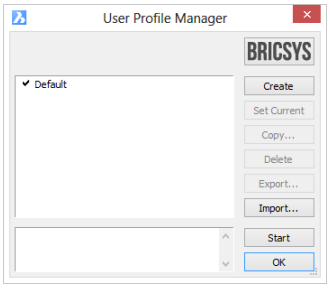
BricsCAD and AutoCAD both support *user profiles* to store each user's customization settings. After changing settings with the BricsCAD **Settings** and AutoCAD **Options** commands, you save the all the settings in a *.arg* user profile file. The idea here is that you can make multiple profiles that 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 V16 | 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 V16\bricscad.exe" /P <UserProfileName>
   ```

   For example, replace <*UserProfileName*> with the *.arg* file's name, such as *myprofile.arg*:

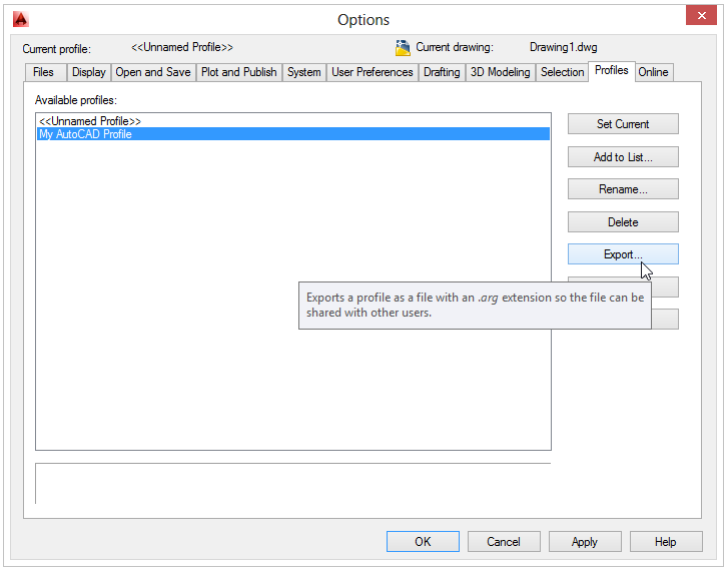
   ```
   "C:\Program Files\Bricsys\BricsCAD V16\bricscad.exe" /P myprofile.arg
   ```

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 2016\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 2016\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\V16x64\en_US*
Mac	
Linux	*home/<login>/Bricsys/BricsCAD/V16x64/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:

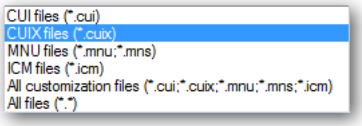
Windows	*C:\Users\<login>\AppData**Roaming**\Bricsys\BricsCAD\V16x64\en_US*
Mac	
Linux	*home/<login>/Bricsys/BricsCAD/V16x64/en_US/*

TIP You can change the path to the local and roamable folders with system variables **LocalRootPrefix** and **RoamableRootPrefix** in BricsCAD.

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 OS X. (They are not hidden in Linux.) Because I access them frequently, I unhide the folders.

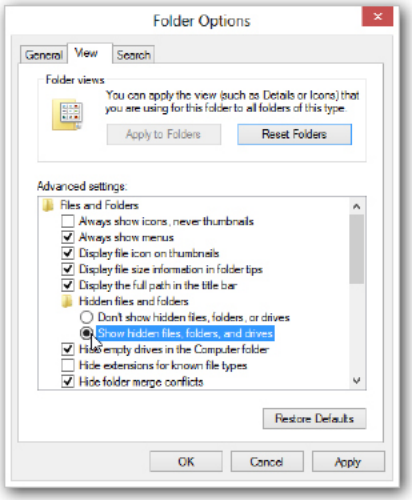
> **TIP** If you find yourself accessing these folders often, create shortcuts on your computer's desktop to them. Here's how: hold down the **Ctrl+Alt** key while dragging the folder name from Explorer onto the desktop.

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

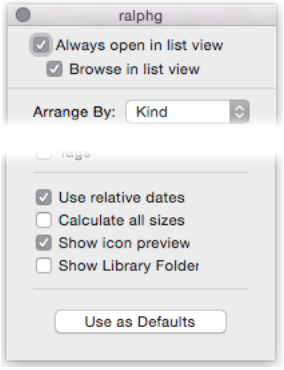
2. Now that hidden folders are visible, follow these steps in Explorer:
   a. Go to the *C:\users\<login>\appdata* folder.
   a. Right-click the folder, and then choose **Properties**.
   b. Uncheck **Hidden**, and then click **OK** to close the dialog box.

You can now see the Local and Roaming folders.

Tutorial: Making Hidden Folders Visible in OS X

The Library folder is where BricsCAD stores its support files on Mac computers. Here is how to reveal the folder in OS X:

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 OS X

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 2016\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\V16x64\en_US\Support*
   - **Mac** */Users/<login>/Library/Preferences/Bricsys/BricsCAD/V16x64/en_US/Support*
   - **Linux** *home/<login>/Bricsys/BricsCAD/V16x64/en_US/support*

   (Remember to replace <login> with your Windows login name.)

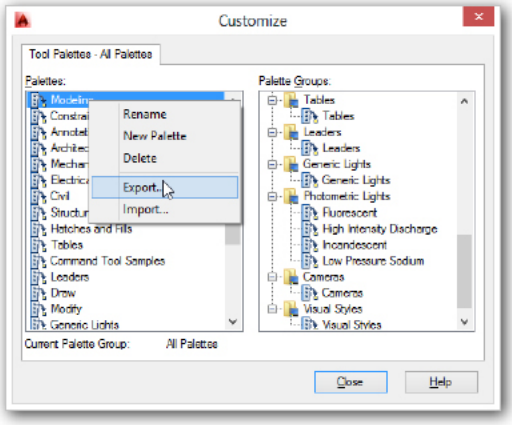
It turns out that in BricsCAD you cannot simply use the Customize dialog box's **Program Parameter File** field, because it does not allow you to enter a different path.

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

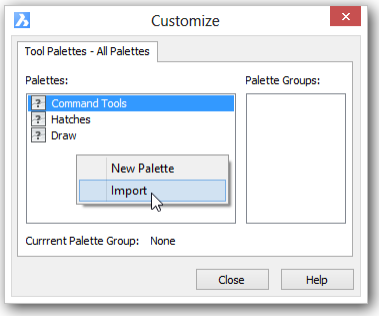
If you have customized the content of AutoCAD's Tools Palette, then you can use them in BricsCAD, because they use the same *.xtp* file format to export and import palettes. (XTP is short for "xml tool palettes," and is a file format based on XML, a self-documenting version of HTML that is often used 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 to export. From the shortcut menu, choose **Export**.



Choosing palettes to export from AutoCAD

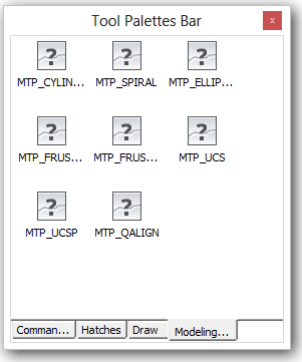
3. In the Export Palettes dialog box, choose the folder in which to place the exported XTP file, and then click **Save**. (I use the Desktop, because it is easy to find!)
4. Switch to BricsCAD.
5. Right-click the Tools Palette bar, and then choose **Customize Palettes**.
6. In the Customize dialog box, 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!

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 OS X, 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 OS X. (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 OS X 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 OS X 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 (www.wxwidgets.org). This interface allows BricsCAD to look the same on Linux, OS X, 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 OS X, 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 OS X.) 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 OS X. 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 OS X; I keep right on working. Even though Apple based OS X on Unix, I am surprised that OS X needs reboots following updates.

Here's a funny thing I have noticed: it is easier to get used to OS X 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 OS X, which is locked to Apple hardware. To try out Linux, you can install it on an existing Windows computer; to try out OS X, you have to buy all new hardware, and get used to different keyboard and trackpad interactions.

(A tip: If you have to get OS X, 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. (OS X 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 OS X, 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, OS X, 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 OS X.

But it cannot run AutoCAD and other powerhouse software found in Windows and OS X, 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.

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 OS X.

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. www.linuxmint.com

Problematic Installers

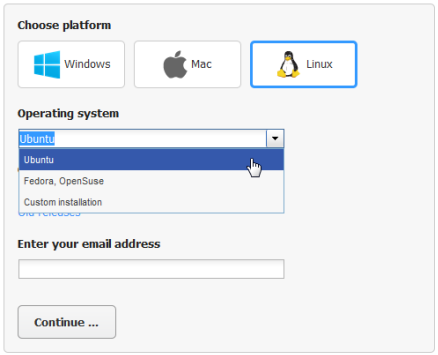
The biggest headache for new and medium-term Linux users is installing software. Many times, installing software goes without a hitch; other times, it does not work well and is a major pain. The problem exists because Linux first expected users to install software through the command-line interface; 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 OS X-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-price AutoCAD. How is this possible? Here are some reasons:

- Autodesk has high operating expenses and 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
- Autodesk considers AutoCAD a platform on which to build extra-cost, discipline-specific add-ons; Bricsys positions BricsCAD Platinum as an all-in-one solution

The result is that BricsCAD will over time gain more functions even as Autodesk pulls back on developing AutoCAD. We see this happening as the more recent releases of AutoCAD were low in new functions, while the new features list for recent releases of BricsCAD go on for pages and 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, and hardware library*
- Surface modeling
- Assembly modeling*
- Kinematic analysis*
- Sections
- Generative drafting (model documentation)
- Bills of material (data extraction)
- BIM modeling*
- Sheet metal design* (add-on module)
- Export-import, including import of assemblies* (add-on module)

*) Functions missing from AutoCAD. Functions shown in **blue** are added to BricsCAD since the last edition of this book.

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 V16 for Windows			BricsCAD for Mac & Linux			AutoCAD	
	Platinum	Pro	Classic	Platinum	Pro	Classic	2016	LT
3D Geometric constraints	●			●				
3D Surfaces	●	●	●	●	●	●	●	
3D Mesh modeling							●	
ACIS modeling and editing	●	●		●	●		●	(1)
ACIS viewing	●	●	●	●	●	●	●	●
Assembly modeling and editing	●			●				
Assembly viewing	●	●		●	●			
Bills of material	●			●			●	
BIM modeling	●							
Deformation modeling	●			●			●	
Design intent	●			●				
Direct modeling	●	●		●	●		●	
Generative drafting	●	●					●	
Hardware library	●	●						
Kinematic analysis	●			●				
Mechanical browser	●			●				
Section planes	●	●		●	●		●	
Surface modeling and lofts	●			●			●	
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

Bricsys initially prefixed the names of all direct modeling command with *dm*, but over time the company is integrating these functions into regular commands. For instance, the Copy and Union commands now work on direct models, and so the dmCopy and dmUnion commands were removed from V15.

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 V16 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	...
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	
Help searches for help topics at the command line	Help
dmSelectEdges places faces and solids in a selection set	...
Ucs locates the UCS icon on entities	DUcs
dmUpdate updates 3D models to satisfy constraints	...

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 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). In the late 1980s, it was the first to popularize history-based parametric modeling. 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.

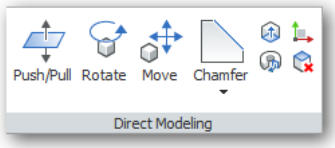
As computers became faster, however, CAD firms were able to implement direct modeling in a more powerful manner, and so it was re-popularized through a new breed of programs, like SpaceClaim and IronCAD. Old software firms like Autodesk and PTC also released new direct modeling software, with New Age 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

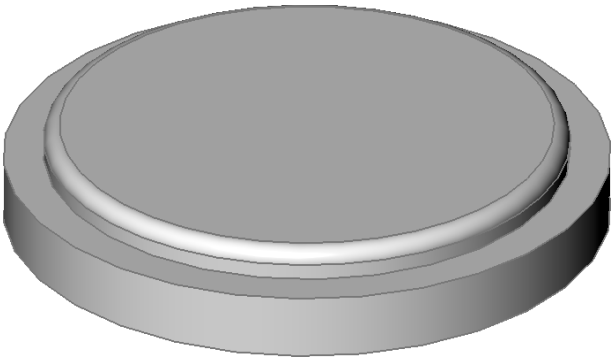
- In the ribbon's **Modeling** tab, look for commands in the **Direct Modeling** panel.



- In the **Model** menu, choose the **Direct Modeling** submenu

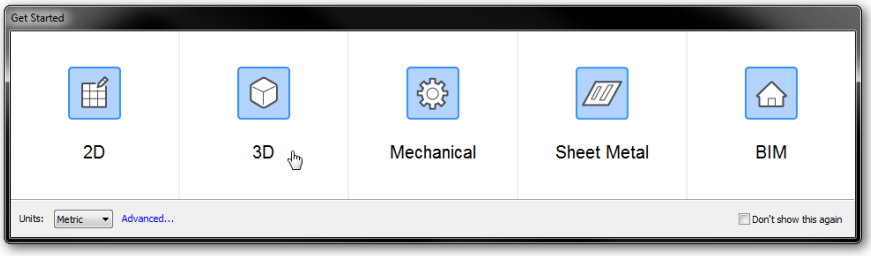
Direct Modeling Tutorial

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

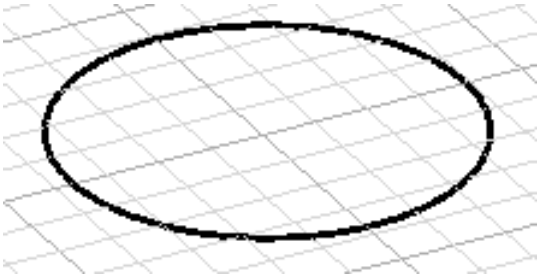
1. Start BricsCAD with a new 3D Modeling drawing in metric mode. (If necessary, switch to **3D Modeling** workspace: right-click the workspace name on the status bar, and then choose "3D Modeling" from the shortcut menu.)



Starting a new drawing in 3D modeling workspace with metric dimensions

2. Draw the base of the lid as a circle 75mm in diameter, as follows:

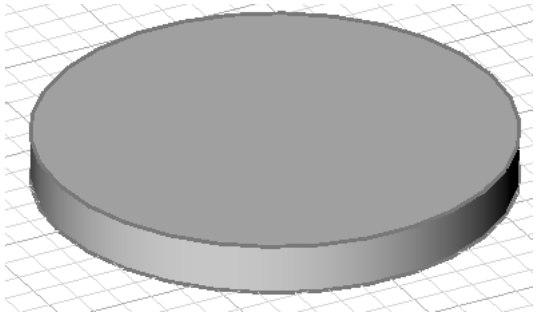
```
: circle
2Point/3Point/TanTanRad/Arc/Multiple/<Center of circle>: (Pick a point in the drawing)
Diameter/<Radius>: d
Diameter of circle: 75
```



Beginning with a circle

3. You will be extruding objects a couple of times in this tutorial, and so I'll show you different ways to do this. For this first extrusion, use the official command. To extrude the circle into a cylinder 16m tall, start the **dmExtrude** command, like this:

```
: dmExtrude
Select planar entities/subentities to extrude: (Select the circle)
Entities/subentities in set: 1
Select planar entities/subentities to extrude: (Press Enter to continue)
Extrude: Limit/<Distance>: 16
```



Extruding the circle to 16mm tall

4. The next step is to add the 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. This time you'll use the dmPushPull command to extrude the circle into a cylinder.

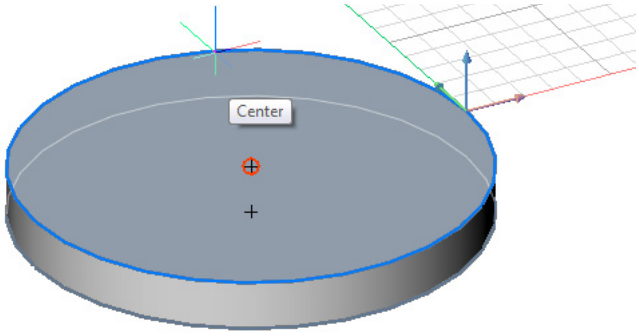
   a. To locate one circle at the precise center of another one, turn on CENter entity snap mode by entering the **Center** command:

   ```
   : center
   ```

   b. Start the **Circle** command.

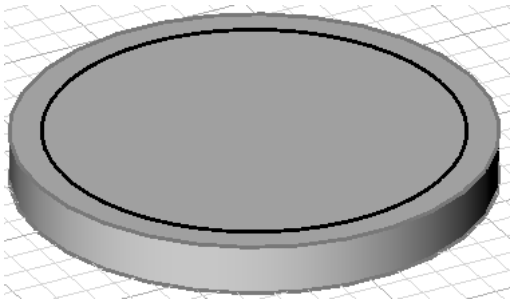
   ```
   : circle
   ```

   c. 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

d. Draw a circle 65mm in diameter.

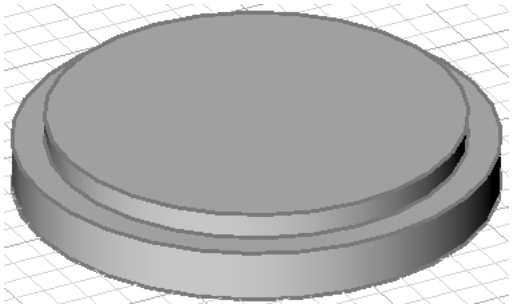


Circle drawn on top of cylinder

e. Start the **dmPushPull** command to pull the circle up by 8mm.

```
: dmPushPull
Select faces or closed 2D contours to push/pull: (Select the new circle)
Entities/subentities in set: 1
Select faces or closed 2D contours to push/pull: (Press Enter to continue)
```

f. Drag the circle up to turn it into a cylinder. Enter **8** for the height, and then press **Enter**.

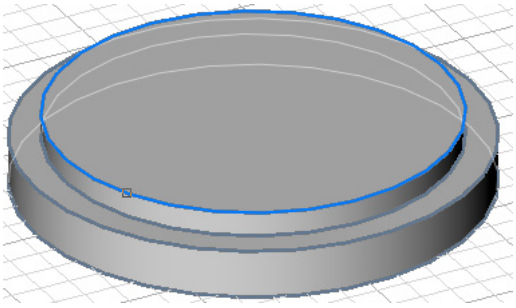


Second cylinder sitting atop the first

5. Round the edges with the **dmFillet** command, as follows:

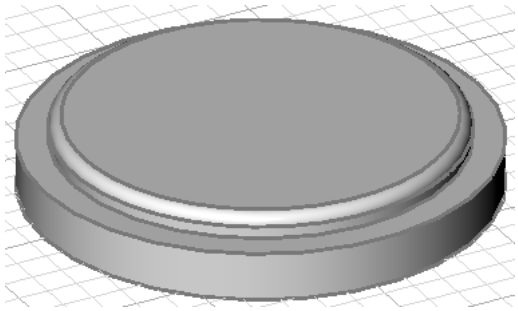
a. Enter the dmFillet command and then chose the edge to fillet:

```
: 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

b. Specify a fillet radius of **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.

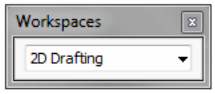
To switch between 2D and 3D drafting environments, BricsCAD uses the same concept of "workspaces" as AutoCAD. BricsCAD V16 comes with the following workspaces; the table compares equivalent workspace names between BricsCAD and AutoCAD:

BricsCAD Workspace Names	Equivalent AutoCAD Workspace Names
2D Drafting	Drafting and Annotation (ribbon)
3D Modeling	3D Modeling (ribbon)
...	3D Basics (ribbon)
Mechanical	...
BIM	...
Sheet Metal	...

V15 removed the 2DContext and 3DContext commands that switched between 2D and 3D interfaces quickly. (new to v16) The Sheetmetal workspace is added to V16.

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

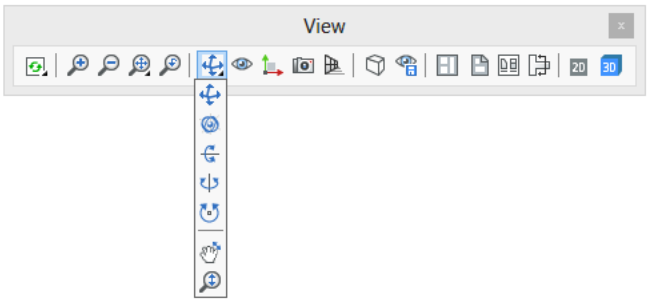
Modeling in 3D is just like drawing in 2D: we still work on a 2D plane for the most part. The 2D plane often is the face of a 3D object. Because 3D objects typically have six or more faces, it is important to move to the correct face quickly. For enable this, BricsCAD has dynamic UCS, which like AutoCAD 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:

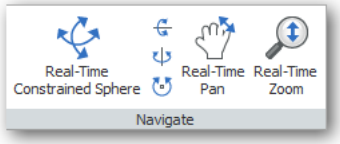
BrisCAD	AutoCAD	Description
RtRot	3DOrbit	Rotates the 3D view dynamically
RtRotCtr	3DCOrbit	Rotates the 3D view about a user-defined center point
RtRotF	3DFOrbit	Rotates the 3D view freely
RtRotX	...	Rotates the 3D view about the screen's x-axis
RtRotY	...	Rotates the 3D view about the screen's y-axis
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



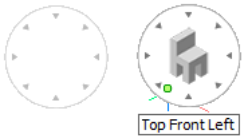
- In the ribbon's **View** tab, choose commands from the **Navigate** tab



- 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.

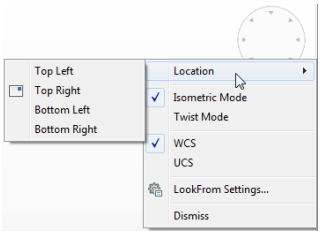


Left: *Lookfrom widget at rest;* ***right:*** *with cursor entering the circle*

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.

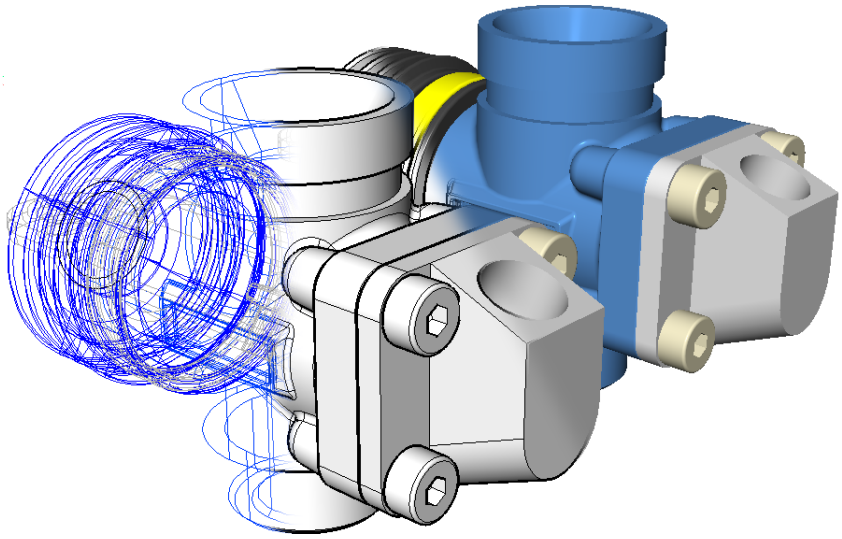
See Chapter 2 “Comparing User Interfaces” for more about the LookFrom widget.

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

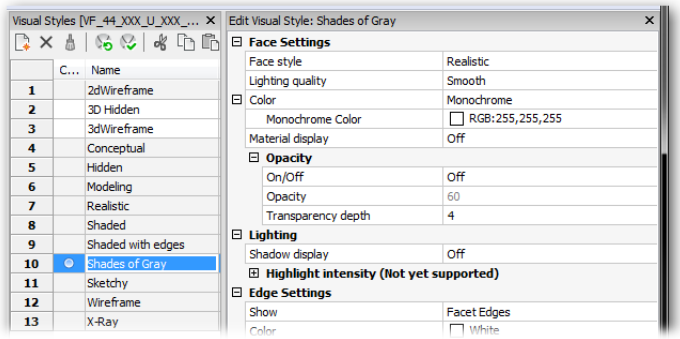
VISUAL STYLES AND RENDERING

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

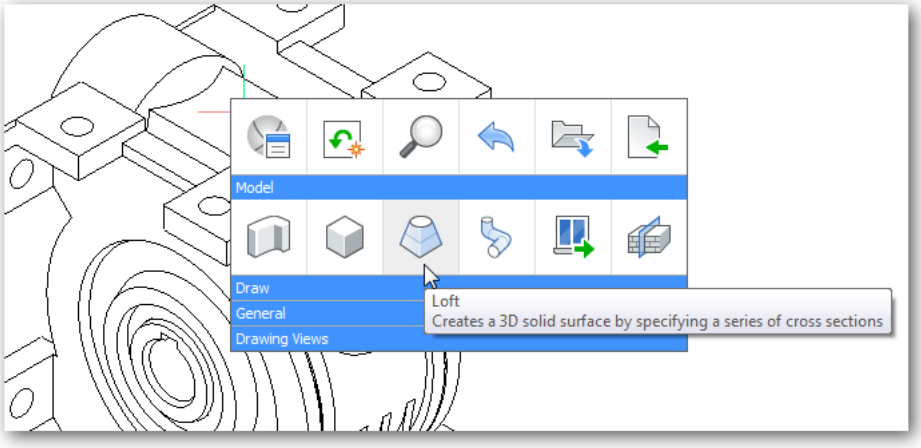
Accessing Visual Styles Commands

- 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. It provides intuitive access to contextual commands. The Quad cursor changes its content, depending on the context. (AutoCAD has nothing similar.)

Contexts that affect the Quad cursor include drawing/editing and the workspace. When no objects are selected, right-click to put the Quad cursor into drawing mode; selecting an object put the cursor into editing mode. The content of the cursor changes, depending on which workspace is current.



Drawing commands displayed by Quad cursor

The Quad cursor is also described in chapters 1 (Introduction) and 4 (Customization) of this book. (NEW TO V16) Quad icons are larger, and drawing mode is new.

TIP When you right-click an empty area of the drawing with QUAD turned off, BricsCAD repeats the last command.

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 certain 3D modeling operations begun with the Quad Cursor. 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*

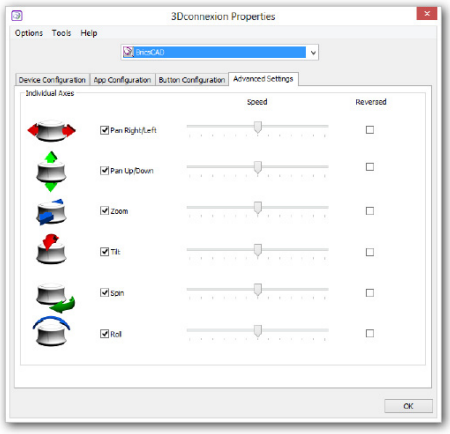
For example, the bar illustrated on the left appears with the **dmExtrude** command. The options displayed are for the Auto, Create, Subtract, and Unite options. Clicking the **x** dismisses the bar; it does not cancel the command. You can still use the Ctrl key to switch between options.

Toggle the display of the Tips bar with the **TIPS** button on the status bar.

WORKING WITH A 3D MOUSE

CAD supports a 3D mouse when it is plugged in and the 3dconnexion driver is installed l running. AutoCAD also supports 3D mice. While AutoCAD provides access to 3D mouse nctions, 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. To access this dialog box in Windows, click the **Start** button, and then choose **All Programs | 3Dconnexion | 3D Mouse Control Panel**, and then click **Properties**.



3Dconnexion control panel determines how BricsCAD reacts to the 3D mouse

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.

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.)

The 3D constraints are available in the Pro and Platinum editions of BricsCAD. The difference is that the Pro version can solve constraints; only the Platinum edition can apply them. Expressions and parameters can specify values and formulae for 3D dimensional constraints, just as AutoCAD does for 2D constraints.

3D Dimensional Constraints
dmAngle3D applies 3D angle constraint
dmDistance3D applies 3D distance constraint
dmRadius3D applies 3D radial constraint

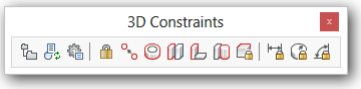
3D Geometric Constraints
dmCoincident3D applies 3D coincident constraint
dmConcentric3D applies 3D concentric constraint
dmConstraint3d is a super command that applies any kind of 3D constraint
dmTangent3D applies 3D tangency constraint
dmFix3d applies 3D fix constraint
dmParallel3D applies 3D parallel constraint
dmPerpendicular3D applies 3D perpendicular constraint
dmTangent3D applies 3D tangency constraint

WORKING WITH 3D CONSTRAINTS

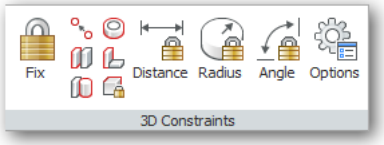
For a tutorial on using 3D constraints, see the Assembly Drawings section later in this chapter.

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 automatically determines what you were probably intending to design. This is known as *design intent*. When design intent is turned on, BricsCAD automatically recognizes parts of 3D entities that ought to be edited together. This is similar to the actions of another MCAD program known as Solid Edge, where the same function is named "Live Rules." AutoCAD does not provide design intent.

For instance, an object has 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 apply design intent to specific ares of a model, unfortunately: design intent is universal. You can choose, however, which aspects of design intent you want operating. For instance, you can have BricsCAD recognize just planes that are 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.

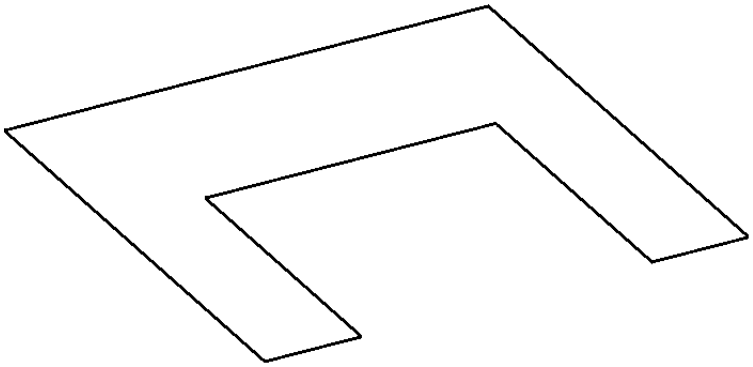
dmRecognize	Description	On by Default
0	All off	
1	Tangent surfaces of planes, cylinders, and cones	
2	Coincident planes	•
4	Parallel planes	•
8	Perpendicular planes	
16	Cylinders perpendicular to planes	
32	Coaxial surfaces of cylinders and cones	•
64	Equal radius on cylinders (or holes) and spheres	•
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

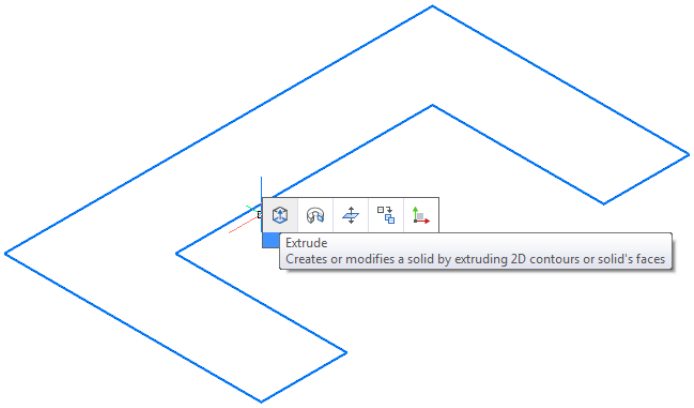
To show how design intent works in BricsCAD, you draw a 3D shape and then use the **dmPushPull** command without — and with — design intent turned on.

1. Start BricsCAD in 3D modeling mode.
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

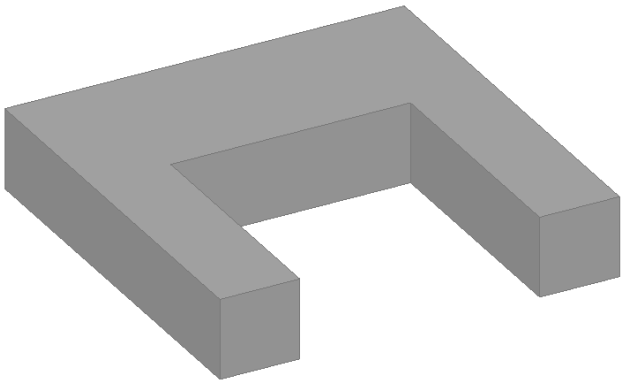
3. Turn the 2D shape into a 3D model by executing the **Extrude** command from the Quad cursor. To do so, follow these steps:
   a. Move the cursor over the polyline. Notice that it turns blue to indicate it is selected by default.
   b. Move the arrow cursor into the Quad cursor icon; notice that it expands.



Exposing the Quad cursor over the polyline

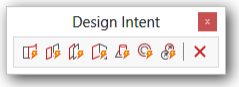
c. Click the **dmExtrude** button. (It is not necessary to select the polyline, a benefit to using the Quad cursor to execute commands.)

```
Specify height of extrusion or [Direction/Path/Taper angle] <1>: 10
```



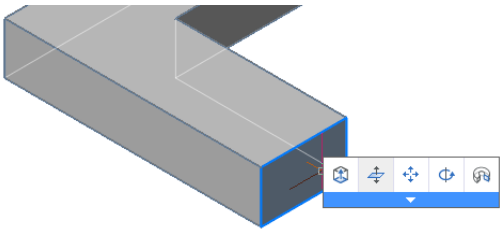
Polyline extruded into a 3D model with the Extrude command

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.



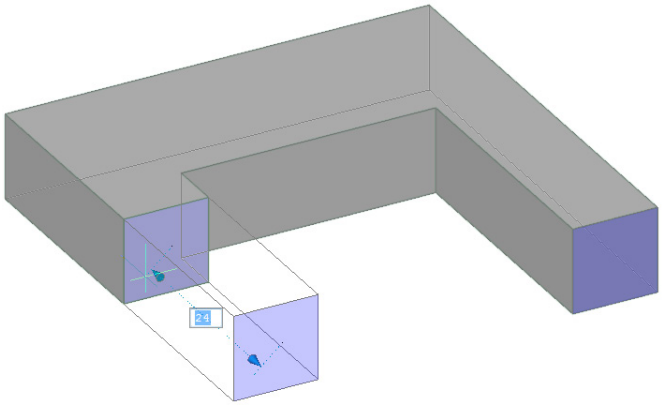
Click the last button on the right to turn off all design intent modes

5. Now you will change the length of one arm with design intent turned off. From the Quad cursor, access the **dmPushPull** command.



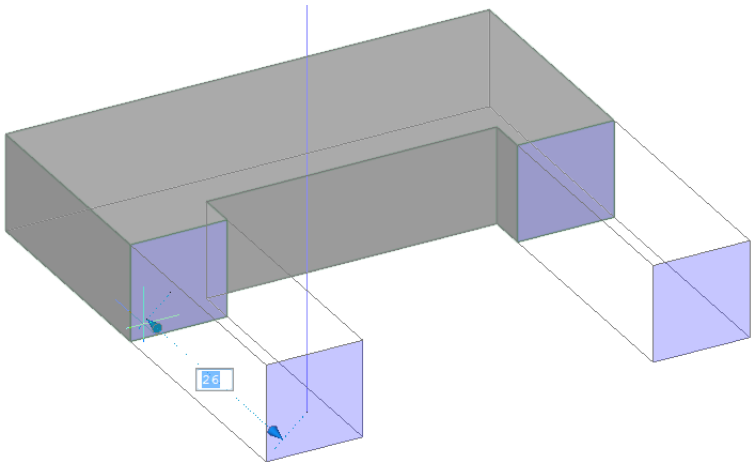
Choosing the dmPushPull command from the Quad cursor

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.

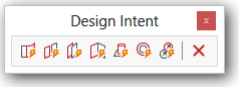


Both planes move together when Coincident Planes is turned on

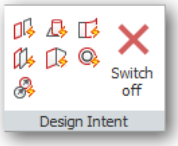
BricsCAD recognized that the other edge was in the same plane as the first one, and so moved it simultaneously and automatically. Should you wish this to not occur, simply turn off design intent.

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.

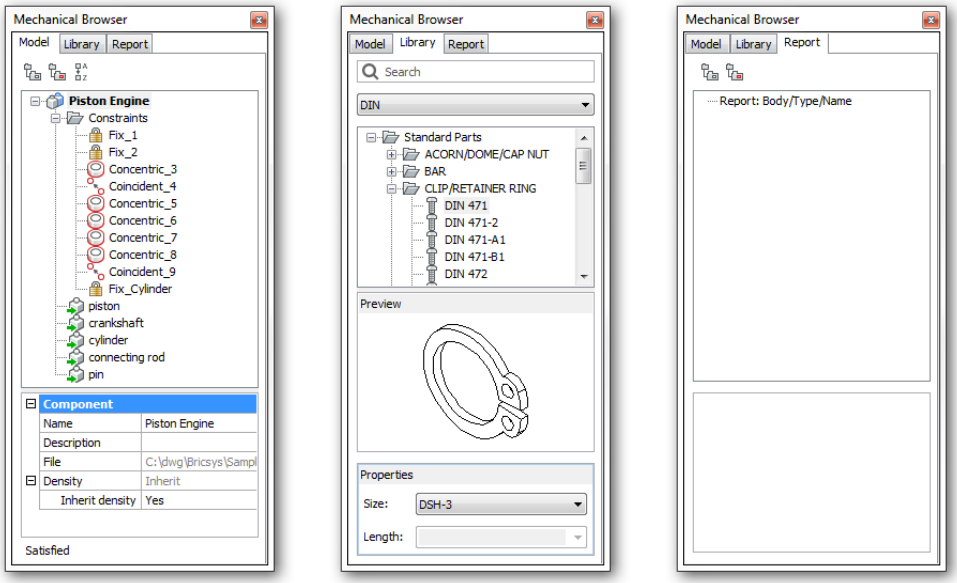


- Enter the **Settings** command and then go to the **Drawing | Drafting | Direct Modeling** section

Mechanical Browser & Hardware Library

BricsCAD Platinum offers the Mechanical Browser bar (palette) that performs 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)
- Model tab lists the constraints that are attached to parts
- Model tab 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)
- (NEW TO V16) 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

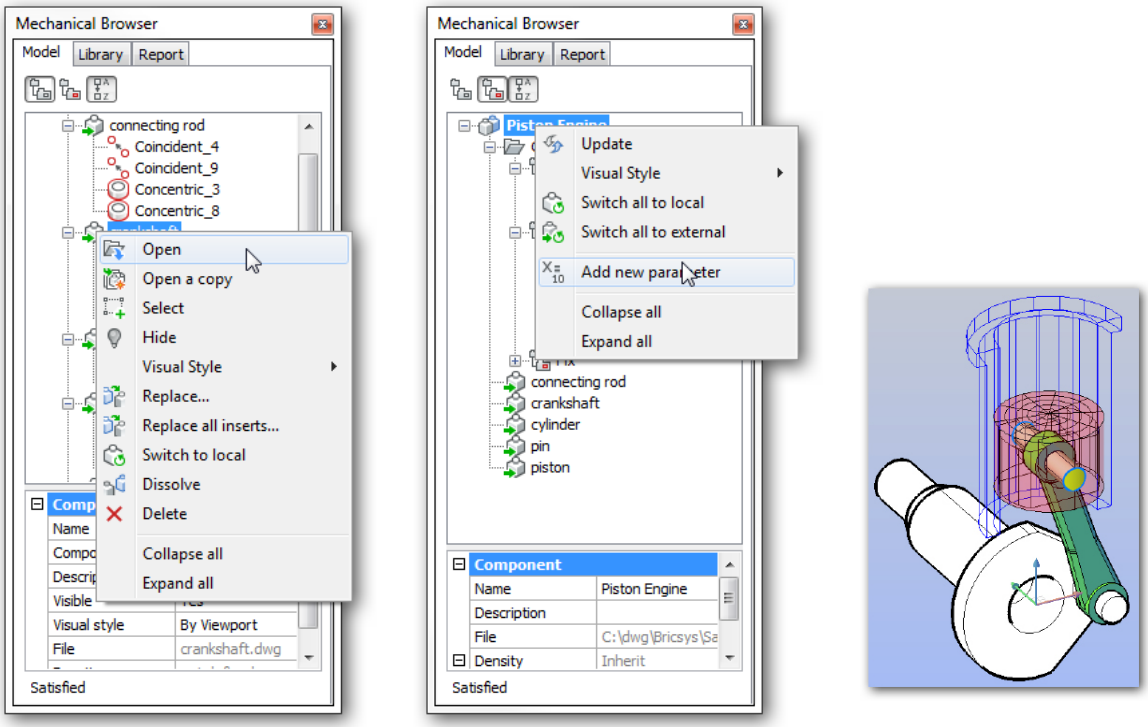
The Mechanical Browser comes into effect when working with 3D models, assemblies, and sheet metal designs. It lists the parts of the models and the constraints used to hold the parts together.

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*

(NEW TO V16) BricsCAD V16 introduces local mechanical components, which are stored in the same *.dwg* file where the component is inserted in the mechanical assembly. Many more commands were added, as shown below.

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

Parts Context Menu

Open opens the part in a new drawing window (bmOpen command)

Open a Copy opens a copy of the selected part in a new drawing window (bmOpenCopy)

Select adds the part to a selection set; repeat to select more parts

Hide turns off display of the part (bmHide command); select again to turn on the part's display (bmShow)

Visual Style specifies the visual style of the part (bmVStyle command); each part can have its own visual style

Replace replaces specific component inserts (bmReplace)

Replace All Inserts replaces all component inserts (bmReplace)

Switch to Local converts external components to local ones (bmLocalize)

Dissolve removes the part from the assembly and reduces it to its constituent entities (bmDissolve command)

Delete erases the part from the drawing (Erase command)

Formula Context Menu

Update reloads all referenced components from external files, updates BOM tables (bmUpdate)

Switch to Local converts external components to local ones (bmLocalize)

Switch to All External converts local components to external ones (bmExternalize)

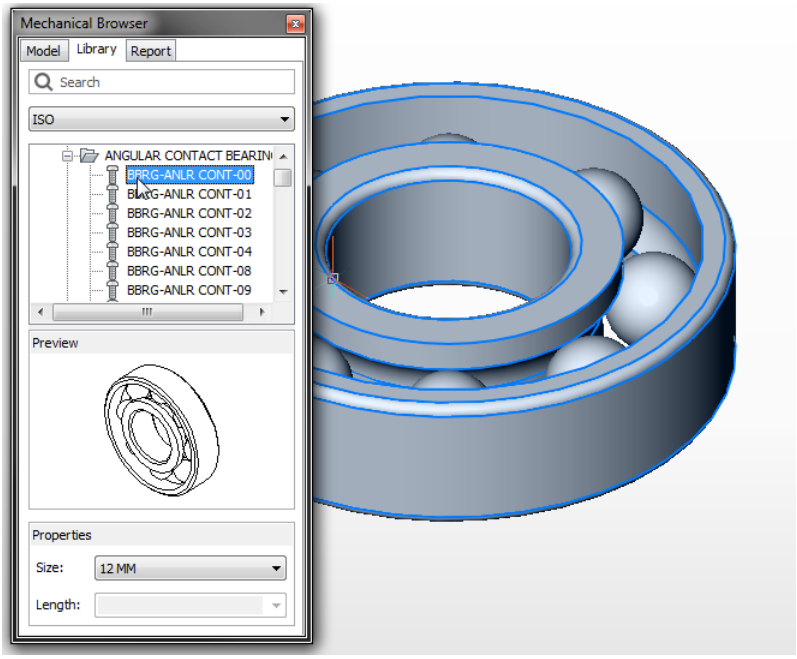
Add New Parameter adds a generic formula (v1=1.0) which is then be edited

Collapse all collapses the tree

Expand all expands all of the nodes of the tree

WORKING WITH THE HARDWARE LIBRARY

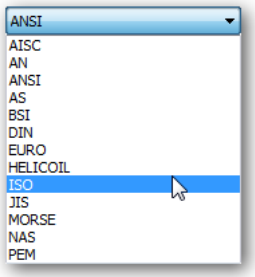
The Hardware Library provides 30,000 parts in parametric form. "Parametric" means that you specify the size of a selected part, and then BricsCAD generates it. AutoCAD does not include a parametric hardware library, but provides access to them online through its Seek command.



Part inserted in the drawing

To pick a part from the library, follow these steps:

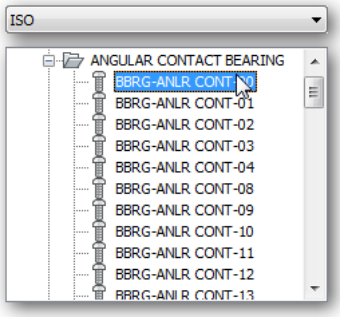
1. From the droplist, choose a standard, such as **EURO**. (The Search field searches only part names within the current standard; it does not search the entire library.)



Choosing an international standard for the parts

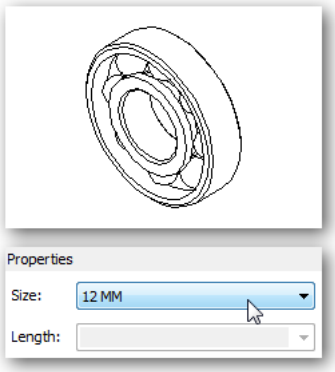
TIP The standard you choose determines the number of parts available. For instance, the JIS standard has the longest list at 47 part types, the ANSI standard has 28 part types, while AN has just one.

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.)



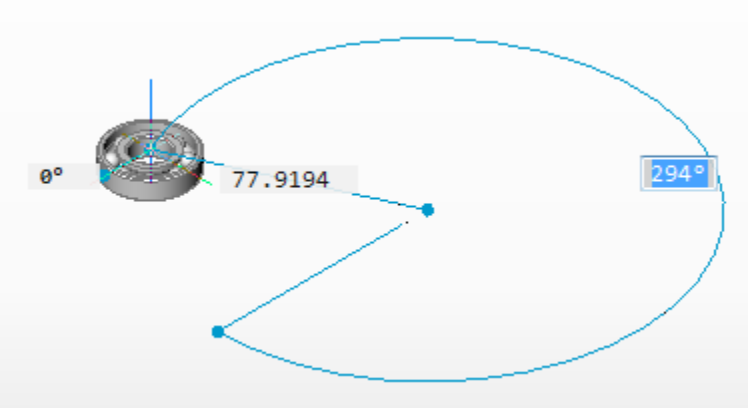
Choosing a part type

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

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

(NEW TO V16) 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. Those shown in **blue** are new since the last release of this ebook; those in **black** have the new ability to work with 3D 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. (NEW TO V16) BricsCAD V16 introduces local mechanical components, which are stored in the same *.dwg* file where the component is inserted in the mechanical assembly. **bmMech** now also converts blocks and external reference to local and external mechanical components. **bmBrowser** add the Report tab to the Mechanical Browser to report the results of dmRepair and smSelectHardEdges commands. Toolbars named "Mechanical" are now named "Assembly."

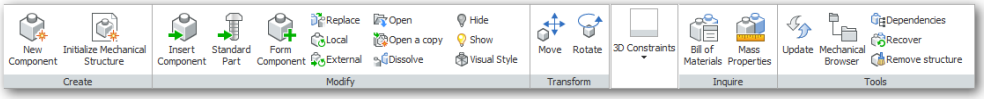
Assembly Modeling Commands
bmDependencies lists names of files containing component definitions in the assembly
bmDissolve dissolves mechanical components inserted into drawings
bmExternalize converts local components to external components
bmForm forms a new mechanical component and inserts it into 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 (previously called "Mechanical")

- In the ribbon's **Assembly** tab, choose a command

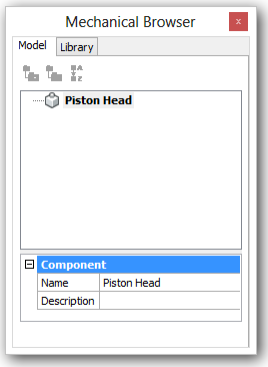


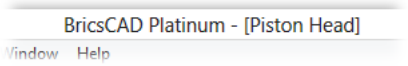
- From the **Assembly** menu, choose a command

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 when you turn a regular drawing into an assembly drawing.

1. Start BricsCAD in the **3D Modeling** workspace.
2. Open the 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**.





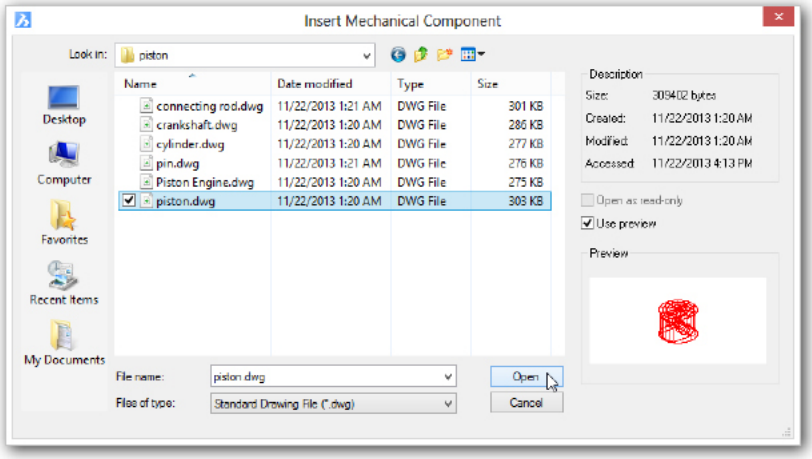
Left: *Drawing renamed by the Mechanical Browser*

Above: *Drawing renamed in the title bar*

   c. Notice that BricsCAD changes the name of the drawing to match. Press **Ctrl+S** to save the drawing.
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.

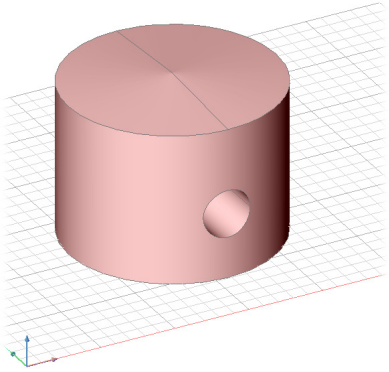


Choosing a DWG file to insert as a component of an assembly

c. Navigate to the *Samples* folder to access mechanical drawings provided with BricsCAD:
`C:\Program Files\Bricsys\BricsCAD V16 en_US\Samples\Mechanical\piston`

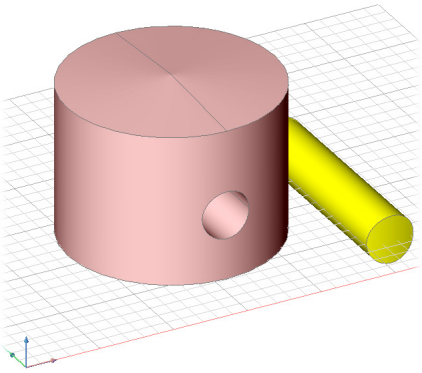
d. Select the *piston.dwg* file and then click **Open**.

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:

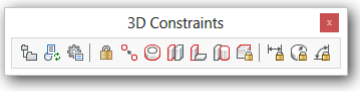
   - You want the pin to stay inside the piston head
   - The pin must be free to rotate, but it cannot slide in and out of the piston

   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**.



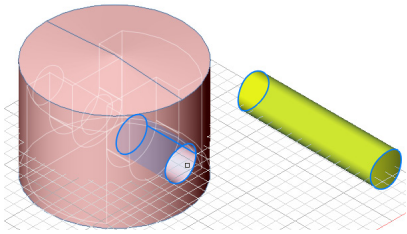
3D Constraints toolbar for attaching components

b. Click **Concentric**:

```
: dmconcentric3d
```

c. And then pick a curved face (a.k.a. subentities) from the piston and the pin:

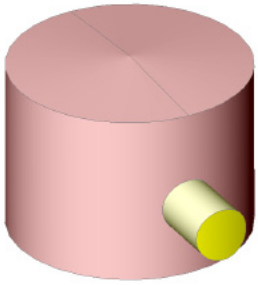
```
Select a pair of subentities: (Pick the curved face of the pin, highlighted in blue on the yellow part shown in the figure below)
Entities/subentities in set: 1
Select a pair of subentities: (Pick the curved inside face of the piston, also highlighted in but on the pink part)
```



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.

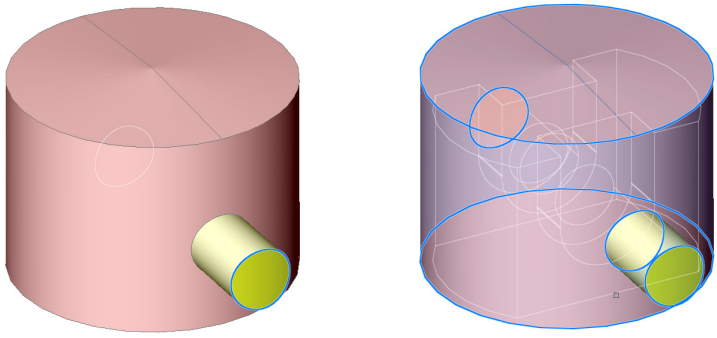
The command ends automatically after you pick the second subentity. Notice that the pin jumps over to the opening of the piston. The pin is inside the piston; now you use the Tangent constraint to keep the pin from sliding out of the piston.



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:

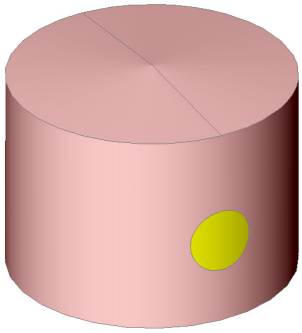
```
: dmTangent3d
Select a pair of subentities: (Pick one end of the pin; see blue outline in the figure below)
```



Left: *Selecting an end of the pin as the first tangent surface;* ***right:*** *Selecting the outside of the piston as the second tangent surface*

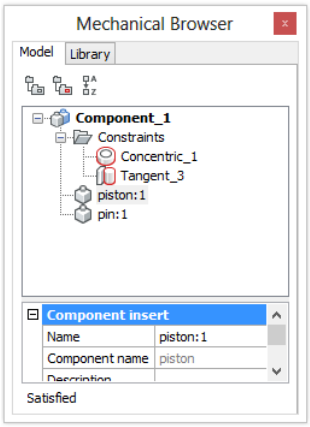
```
Entities/subentities in set: 1
Select a pair of subentities: (Pick the outside of the piston, shown outlined in blue in the figure above)
```

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. 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.

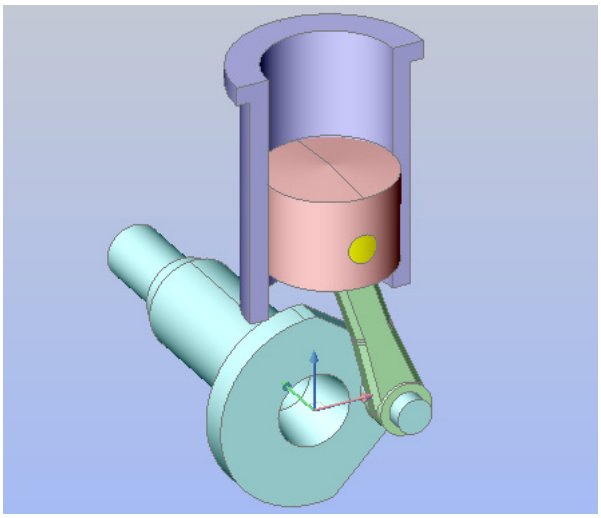
3D Kinematic Analysis

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

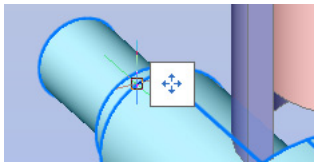
To see how kinematic analysis works in BricsCAD, open *Piston Engine.dwg*, a sample drawing provided with BricsCAD. (You'll find it in the *C:\Program Files\Bricsys\BricsCAD V16 en_US\Samples\Mechanical\Piston* folder.) This assembly drawing is complete, with all of the components held in place with 3D constraints. See figure below.



Sample drawing provided with BricsCAD

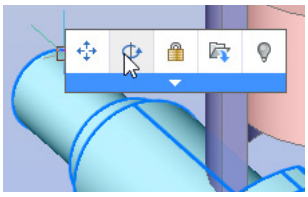
In this sample drawing, you rotate the parts of the mechanism with the **dmRotate** command. Start the command with the Quad cursor, like this:

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.

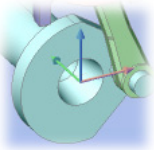


Selecting the Rotate command from the expanded Quad cursor

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.

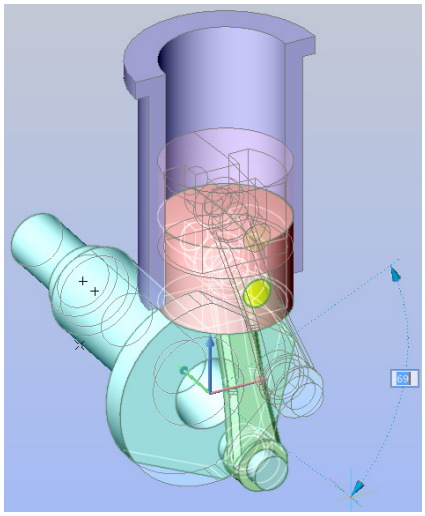


Crankshaft's centerline laying along the y axis

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.

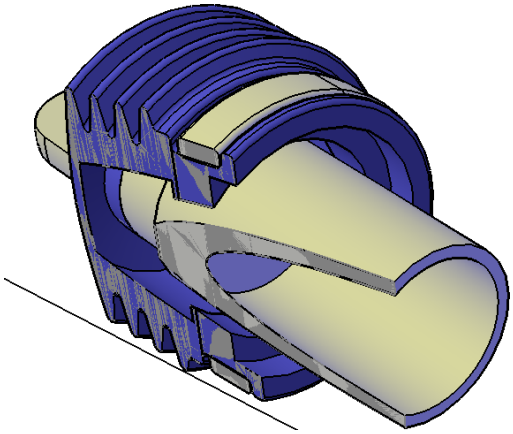
3D Sections

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.

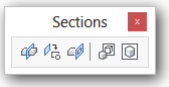


Live sectioning a 3D model in BricsCAD

Accessing the Commands

To access this 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

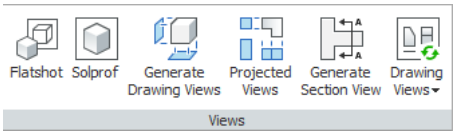
BricsCAD Pro and Platinum editions generate 2D drawings and sections from 3D models. These are called "drawing views" (or "generative drawings" in earlier releases). Because the drawings are associative, they update automatically when you make changes to the 3D model. AutoCAD has the same function, but calls it "model documentation." Commands new since the last edition of the book are shown in blue.

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

Accessing Generative Drawing Commands

- Enter the one of the commands listed above
- Open the **Drawing Views** toolbar

- In the ribbon's Modeling tab, select commands from the **Views** panel



- From the **View** menu, open the **Drawing Views** submenu

GENERATING DRAWINGS FROM MODELS

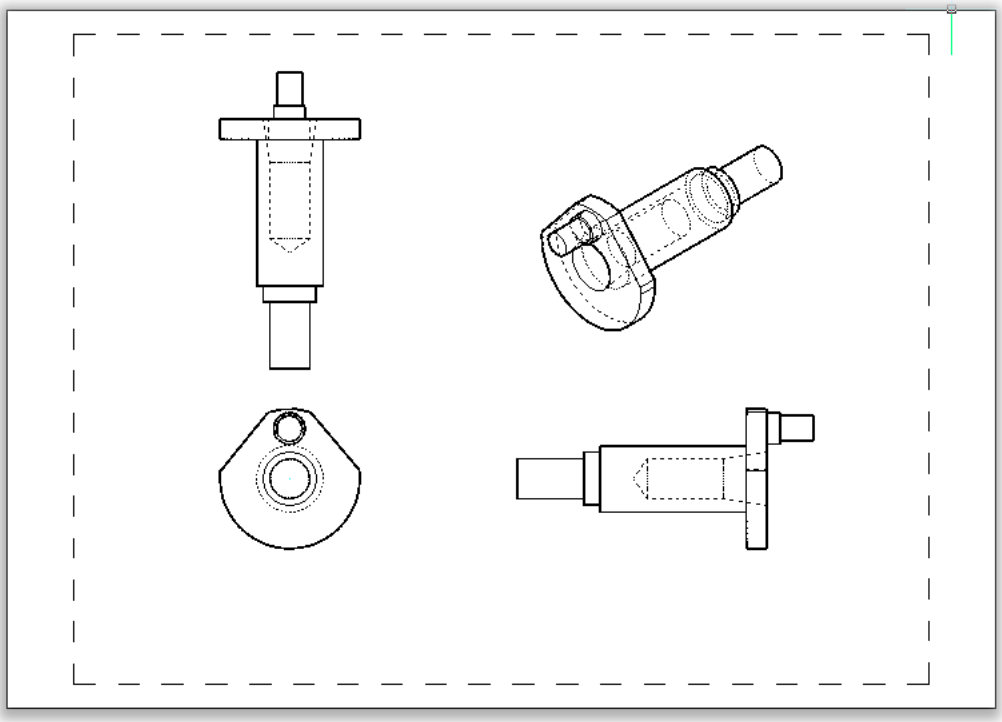
The method of placing 2D views of 3D models in BricsCAD is similar to that of AutoCAD:

```
: ViewBase
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 main view [Scale/Tangent edges/Orientation/Projection type/Exit] <Exit>:
(Pick a point to locate the first view, which is the front view)
Select position for current view [Exit] <Exit>: (Keep picking locations for views...)
Select position for current view [Exit] <Exit>: (...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)
```

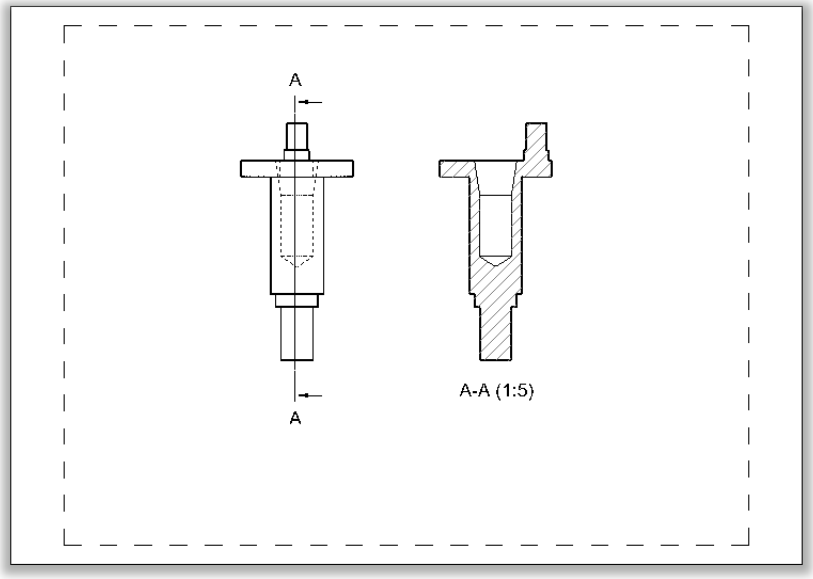
Now pick two points to become the start and end of the section line (A-A) that bisect the view:

```
Specify start point of section line: (Pick a point at one end of the view)
Specify end point of section line: (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.



Section view created by BricsCAD

Details from Drawings

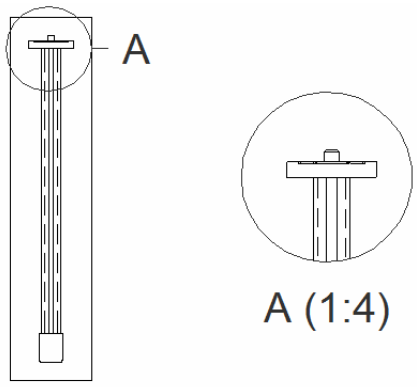
BricsCAD creates detail views through the **ViewDetail** command, as follows:

```
: 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/Exit] <Exit>: (Pick a point to locate the detail.)
```

The default scale for the detail view is 1:4 (four times larger). Enter **S** to change the scale factor:

```
Adjust view scale [Standard scales/Custom/Relative custom/Exit] <Standard scales>
```

Standard scales are those provided by the ScaleListEdit command. You cannot make detail views from detail views.



Detail view A

Section and Detail Styles

(NEW TO V16) 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.

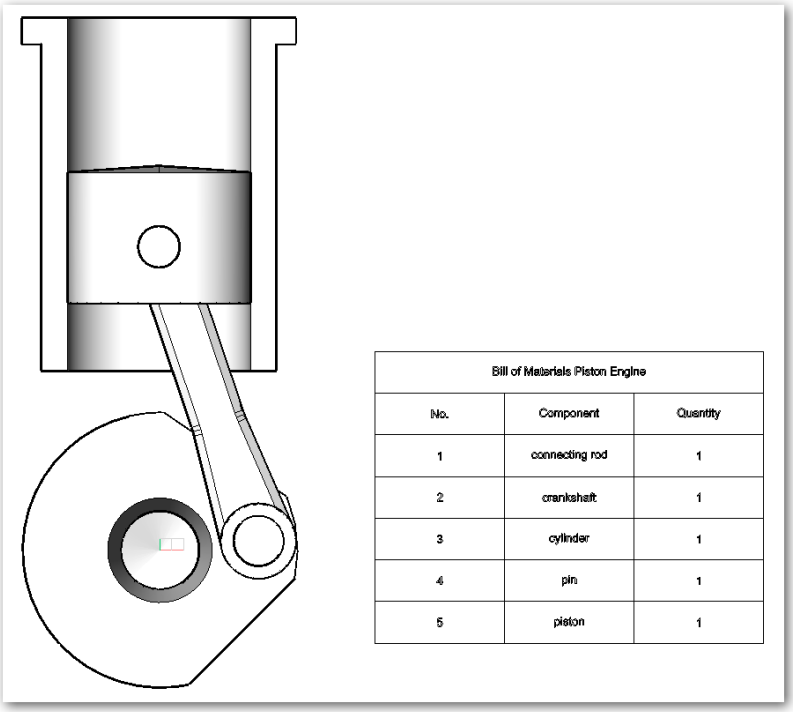
Bills of Material vs Data Extraction

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)
```



BOM table inserted in a drawing

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.

BricsCAD V15 added new optional columns for assembly bills of materials tables: Density, Description, Volume, Mass, and Parameters.

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

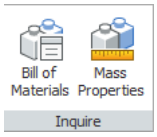
Bill of Materials Piston Engine		
No.	Component	Quantity
1	connecting rod	1
2	crankshaft	1
3	cylinder	1
4	pin	1
5	piston	1

Elements of a bill of materials

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 Sheet Metal Design

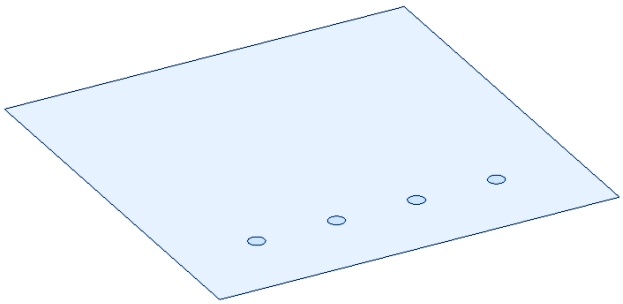
BricsCAD Platinum creates, bends, and unbends sheet metal designs with the optional, extra-cost Sheet Metal add-on.

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
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
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, rebuilds them tangent to adjacent flanges
smSelectHardEdges selects all hard edges, and then reports about them in the report panel
smRethicken restores 3D solid models from sheet metal part by thickening one side
smUnfold unfolds sheet metal bends

TUTORIAL I: HOW SHEET METAL DESIGN WORKS

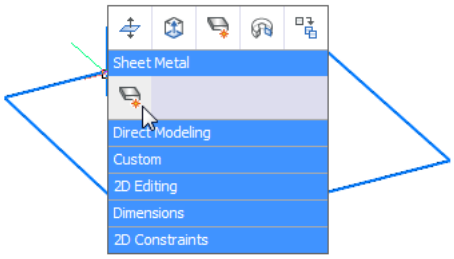
Sheet metal design begins with 2D profiles or 3D models, includes those imported into BricsCAD from other MCAD systems. This tutorial takes you through the fundamental steps using a 2D profile:

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



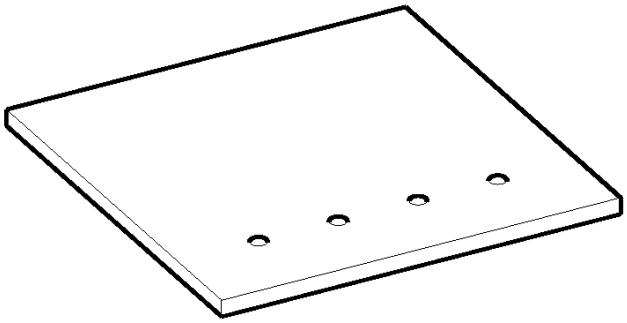
Rectangle and four circles converted to a region entity

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 is thickened. The object is a 3D solid that BricsCAD recognizes as a sheet metal object.

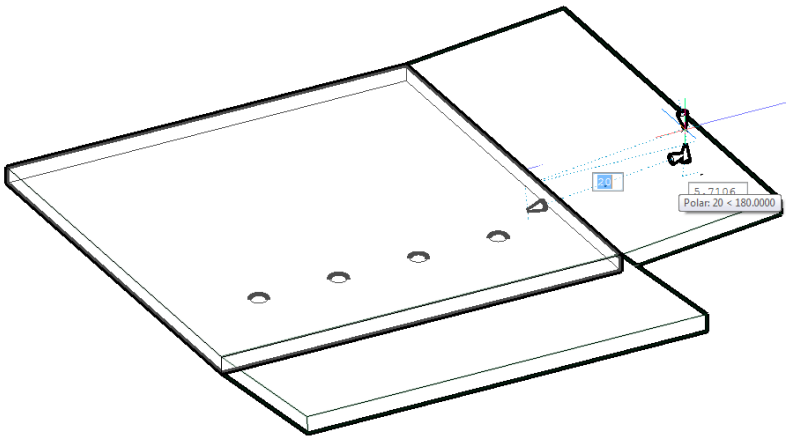


The smFlangeBase command thickens the region

4. To create sides (*flanges* that are pulled from the base), apply the **smFlangeEdge** command:

```
: smFlangeEdge
Select one or more edges of existing flanges: (Pick an edge)
Entities/subentities in set: 1
Select one or more edges of existing flanges: (Pick an adjacent edge)
Entities/subentities in set: 2
Select one or more edges of existing 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.

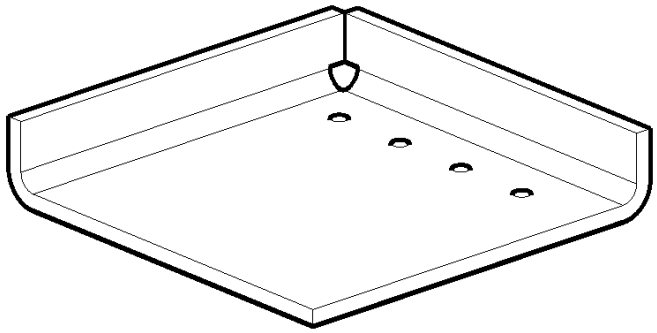


Two edges selected to bend

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 wall [Angle/Length]: (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)
```

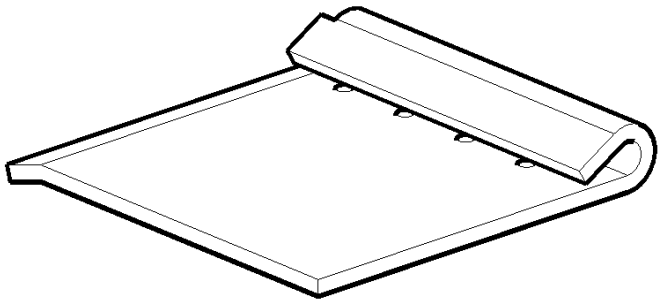
Notice that this command adds bends, bend reliefs, and corner reliefs automatically.



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 rebent, 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)
```

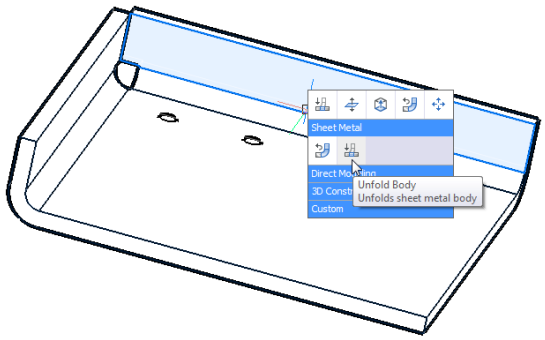


Changing the angle of flanges

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.

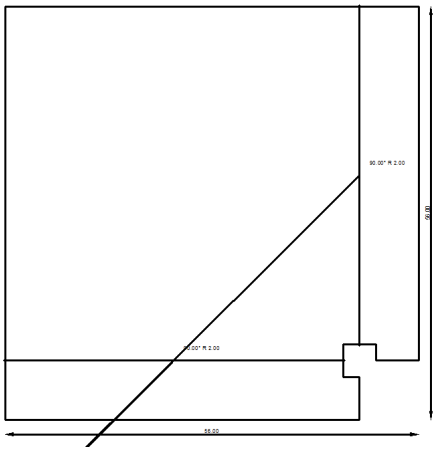
7. Designs are unfolded with the **smUnfold** command. The command is like the flatten command of other sheet metal programs. This command performs two jobs: it generates a 2D drawing of the sheet metal part, and then optionally exports the drawing in DXF format for use with CAM (computer-aided manufacturing) systems of sheet metal parts.

Start the command from the Quad cursor:



Accessing the smUnfold command

```
: smUnfold
Place unfolded body: (Pick a point to place the 2D drawing)
Validate the unfolded body and select an option [save 2D geometry/save 3D geometry/export to
Jetcam/Keep] <Keep>: (Enter an option; see table below)
```

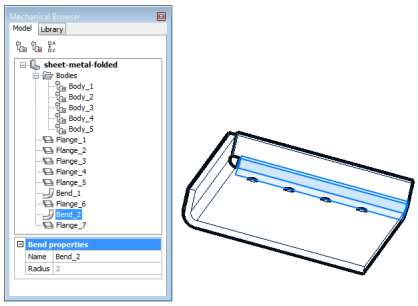


Annotated 2D drawing of the sheet metal part

The options of the smUnfold command have the following meaning:

smUnfold Option	Description
Select flange face or lofted bend face to start unfolding from	Pick the base of the sheet metal body
save 2D geometry	Saves the 2D geometry of the unfolded metal part as a DWG or DXF file
save 3D geometry	Saves the 3D solid of the unfolded metal part as a DWG or DXF file
export to Jetcam	Saves the 2D geometry of the unfolded metal part, assigning layers based on Jetcam standards, and creating a separate *drawing_name*.ppi file for unit and material specifications: #UNITS=MM #MATERIAL=ST33 #THICKNESS=2.000000
Keep	Places the unfolded 3D solid in the current drawing

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

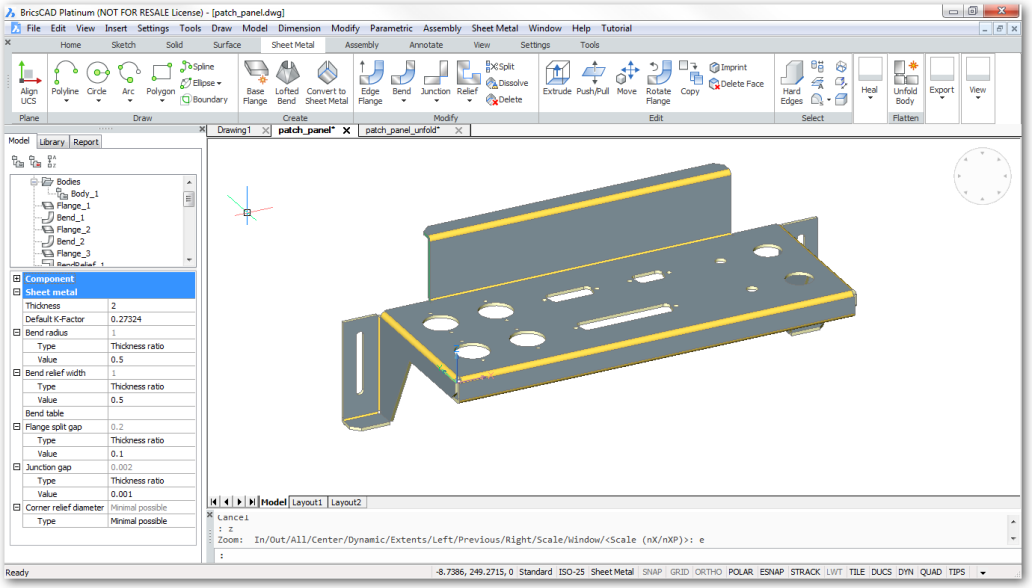
Applying Bend Tables

You can specify the bend radius for every bend individually, or else apply a global radius. In addition, BricsCAD can read in bend tables. These tasks are performed in the Mechanical browser, Model tab.

	A	B	C	D
1	BricsCAD			
2	Version	1		
3	AngleType	Internal		
4	LengthType	BendDeductionTangent		
5				
6	Thickness	0.8		
7	AngleValues		90	
8	Radius	0.8	1.65	
9				
10	Thickness	0.8		
11	DieWidth	6		
12	AngleValues		90	
13	Radius	0.8	1.65	
14				

Bend tables displayed by a spreadsheet program

BricsCAD includes sample bend tables in *C:\Program Files\Bricsys\BricsCAD V16 en_US\Samples\Mechanical\bend_tables*. These are CSV files (comma separated value) that can be opened in a spreadsheet for editing. See the figure below for an example, this one from the *mild_steel.csv* file.

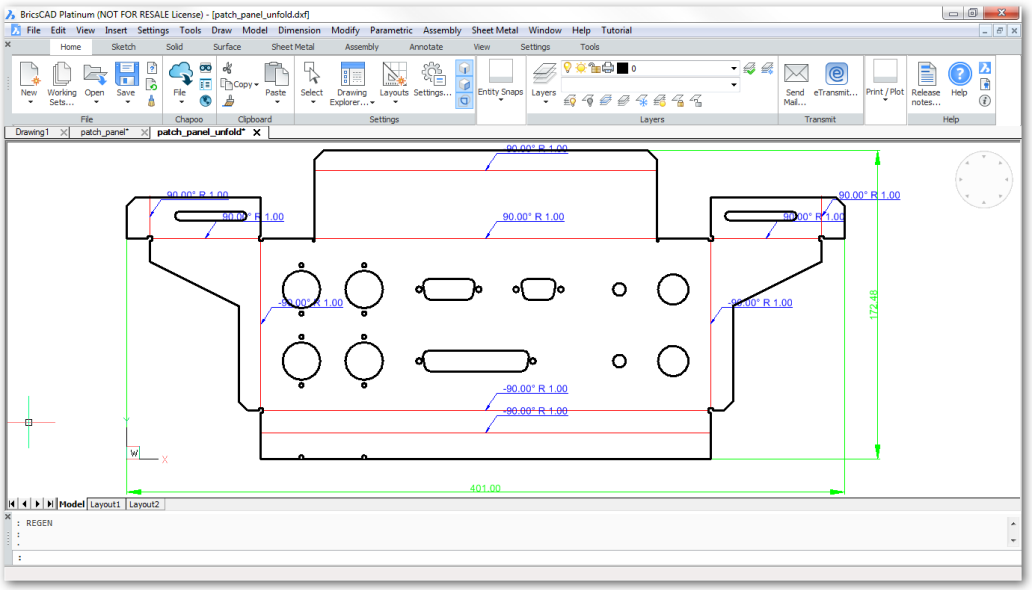


Choosing a bend table for a sheet metal part

To add a bend table, you go into the Model tab of Mechanical Browser, and then choose the name of the part. (See figure below.) Open the **Bend Relief Width** section, and then in the **Bend Table** section, click the Browse button. Choose a CSV file from the *C:\Program Files\Bricsys\BricsCAD V16 en_US\Samples\Mechanical\bend_tables* folder, and then click **Open**.

Exporting Sheet Metal Parts

The **smUnfold** command flattens (unfolds) 3D sheet metal parts into 2D. See figure below.



Sheet metal part unfolded and automatically dimensioned

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.

(NEW TO V16) The smExportToOsm command is removed from V16, and replaced with the **smTargetCXAM** system variable to specify which CAM system to export to.

TUTORIAL II: FROM 3D SOLID TO SHEET METAL

The tutorial above 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 creates sheet metal models from 3D solids, which is a better approach for complex parts. This tutorial shows you how to accomplish this.

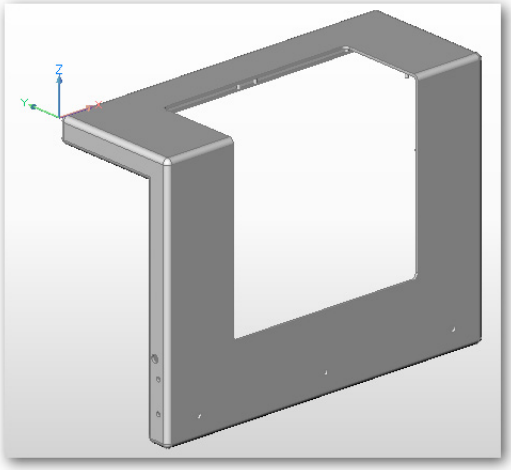
(NEW TO V16) MCAD programs like Solid Edge and Solidworks also have the ability to convert 3D solids into sheet metal parts. In this area, BricsCAD has a distinguishing feature, because the other two MCAD programs make the same mistake: the basic feature is an inseparable flange+bend, whereas in BricsCAD 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.

1. Start BricsCAD in the **Sheet Metal** workspace, and then open the sample file *startfromsolid.dwg*.

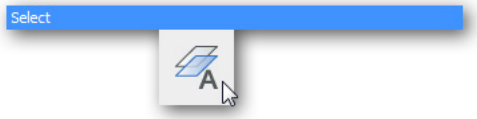


Solid model with pins and filleted corners

2. Defeaturing is done with two commands, 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 pin, and it selects all other identical faces in the mode. Then you remove the pins by subtracting them with the direct modeling version of the Extrude command, dmExtrude.

Here are the steps involved. While you can perform smart selection at the command prompt, is it so much easier using one of the icons:

- From the Quad cursor, choose **Select** > **Same Area Faces**

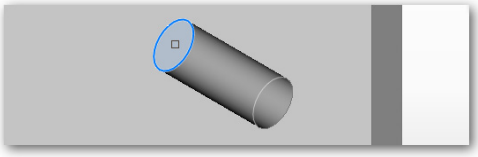


- In the Sheet Metal ribbon's **Select** panel, click the **Same Area Faces** button

3. Ignore the plural nature of the prompt by selecting the face of just one pin:

```
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
```

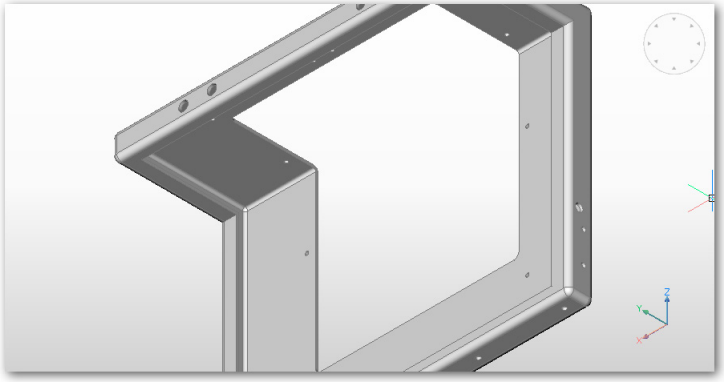
b. Specify 's' for the Subtract option:

```
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.)
```

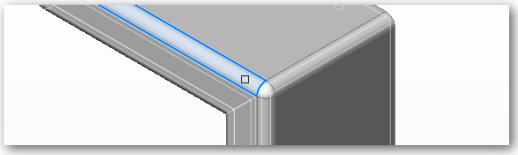
Notice that all of the pins in the model disappear instantly. They are replaced by holes, which will be stamped during the sheet metal manufacturing operation, after which pins are added separately.



Pins removed from solid model

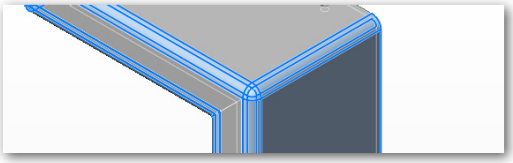
5. The other preparatory step is to remove the fillets so that the edges can later be turned into bends. Again, it is a two-step process: first select all fillets with dmSelect, and then erase them with the dmDelete command.

a. From the Sheet Metal ribbon's **Select** panel, choose the **Same or Less Radius Fillets** icon.

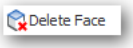


Selecting a fillet

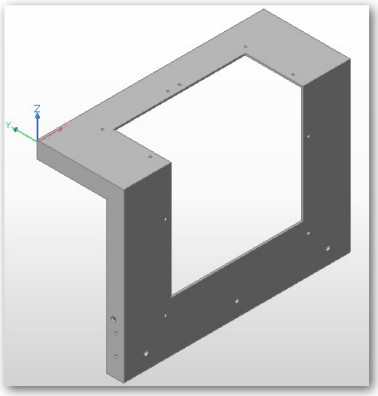
b. Choose a fillet. Notice that BricsCAD selects all the other fillets on the model, as shown in blue in the figure below.



All fillets selected in the model

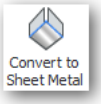


c. From the Sheet Metal ribbon's **Edit** panel, click **Delete Face**. Notice that all corners become sharp.



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.



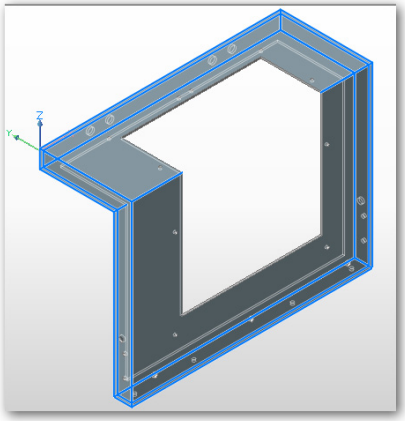
From the Sheet Metal ribbon's **Create** panel, choose **Convert to Sheet Metal**.

```
: smConvert
Select 3D solids/<Entire model>: (Press Enter to select the entire model)
```

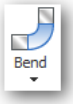
At the prompt, pressing **Enter** selects the entire model. The model looks no different, except that it takes on a gray color. From now on you edit it with commands that start with 'sm', short for sheet metal.

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:

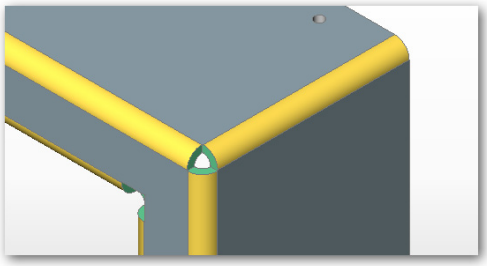
a. From the Sheet Metal ribbon's **Select** panel, click on **Hard Edges**. Notice that all hard edges are selected by BricsCAD, because they turn blue.



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.

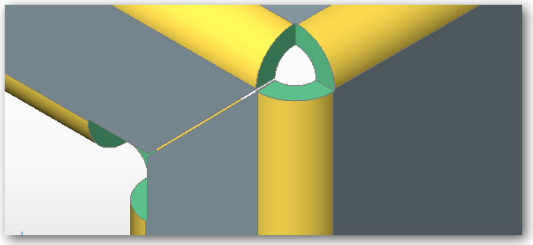


Bends (in yellow) complete with cutouts (in green) at intersections

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 the 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)
```



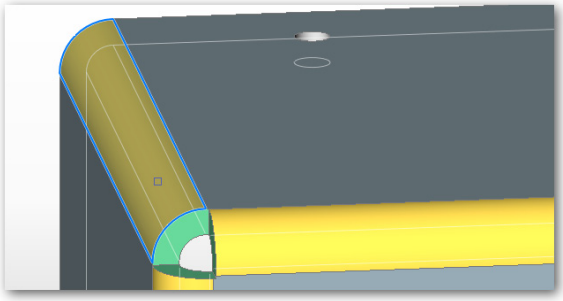
Splitting a flange

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:

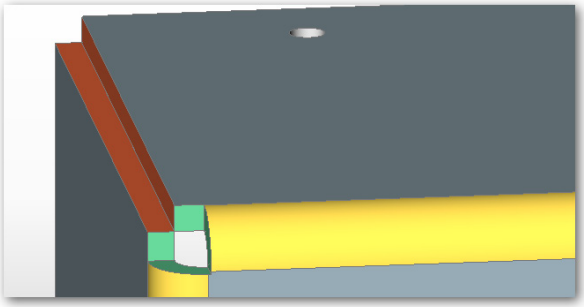
a. From the Sheet Metal ribbon's **Modify** panel, click **Junction**.

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.



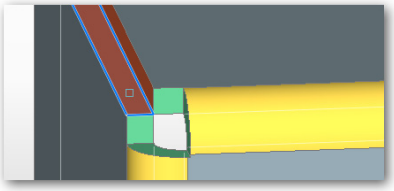
...and turning it into a junction (shown in red)

10. The junction needs to be edited so that one edge cleanly meets the other. You do this with the smJunctionSwitch 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)
```

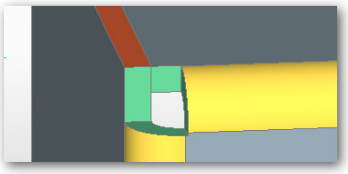


Selecting a face (in red)...

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.

THE COLOR OF SHEET METAL

BricsCAD uses a color coding system to identify features in sheet metal parts. The colors are listed below. Bends are shown in yellow, for example, while corner reliefs (openings) are bright green.

Color of the contour layer	White
Color of the bend lines layer	Red
Color of the bend annotations layer	Blue
Color of the overall dimensions annotations layer	Green

Flange feature color	(144,164,174)
Bend feature color	(255,220,80)
Lofted bend feature color	(160,220,250)
Bend relief feature color	(100,210,150)
Corner relief feature color	(100,210,150)
Junction feature color	(255,110,64)

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."

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.

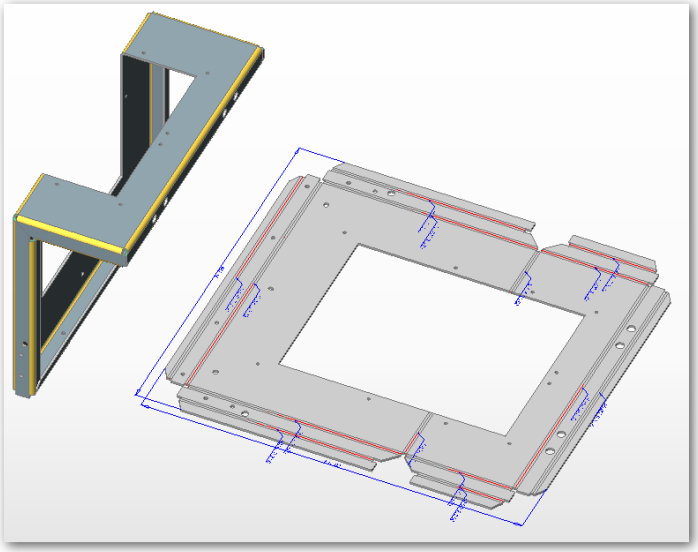
a. From the Sheet Metal ribbon's **Flatten** panel, choose the **Unfold Body** button. At the prompt, just pick any point on the sheet metal body:

```
: 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.

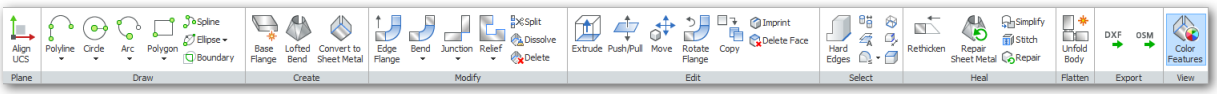


3D model flattened, ready for export to CNC machinery

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

3D BIM Design

BricsCAD Platinum adds commands for modeling buildings in 3D and then exports them as IFC files. Any 3D solid can be used in 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.

(NEW TO V16) The BIM database now allows custom material and composition properties to be defined through the **bimAttachComposition** command. To replace windows and doors, you now simply assign a different definition file using the File property in the properties panel. The **bimDrag** is updated to dynamically place parallel copies when holding down the **Ctrl** key while dragging a major face of a wall and slab solids. The **bimInsert** command now switches between width and height of the door, and distance to neighbor-entities when holding down the **Ctrl** key. When hovering the cursor over an existing door or window, the command places windows and doors with the same parameters.

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
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
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 imprints 2D window and door outlines into walls
bimWindowUpdate updates openings made by windows and doors when their definition changes

HOW BIM DESIGN WORKS

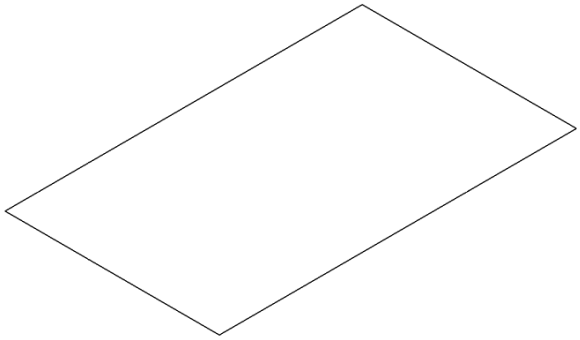
BIM designs commonly begin with 2D floor plans, which are extruded with the **PolySolid** command into walls and floors.

1. Start BricsCAD with the **BIM** workspace and **Imperial** units.
2. To make it easier to see your work, change the visual style from **BIM** to **Wireframe**. You can do this in the Drawing Explorer with the **VisualStyles** command, or else 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] <Wireframe>: 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
Chamfer/Elevation/Fillet/Rotated/Square/Thickness/Width/Area/Dimensions/<Select first corner
of rectangle>: d
Length to use for rectangles <0">: 50'
Width to use for rectangles <0">: 25'
Chamfer/Elevation/Fillet/Rotated/Square/Thickness/Width/Area/Dimensions/<Select first corner
of rectangle>: 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 = 10', Width = 10", Justification = Center, Separate solids = On,
Dynamic = 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

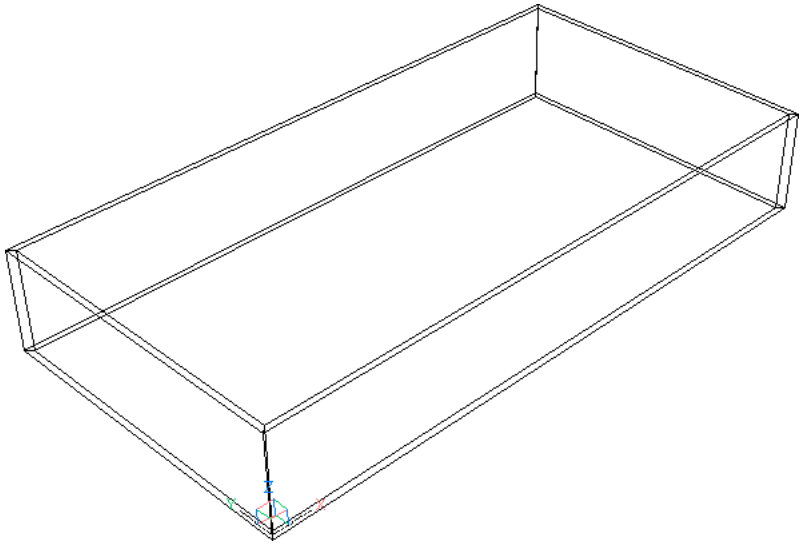
```
Start point or Height/Width/Justification/separateSolids/Dynamic/<Entity>: d
Dynamic height On/OFF <On>: off
Start point or Height/Width/Justification/separateSolids/Dynamic/<Entity>: w
Width of polysolid <10">: 6"
Start point or Height/Width/Justification/separateSolids/Dynamic/<Entity>: h
Height of polysolid <10'>: 8'
```

(NEW TO V16) The name of the "Object" prompt is changed to **Entity**.

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/separateSolids/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.

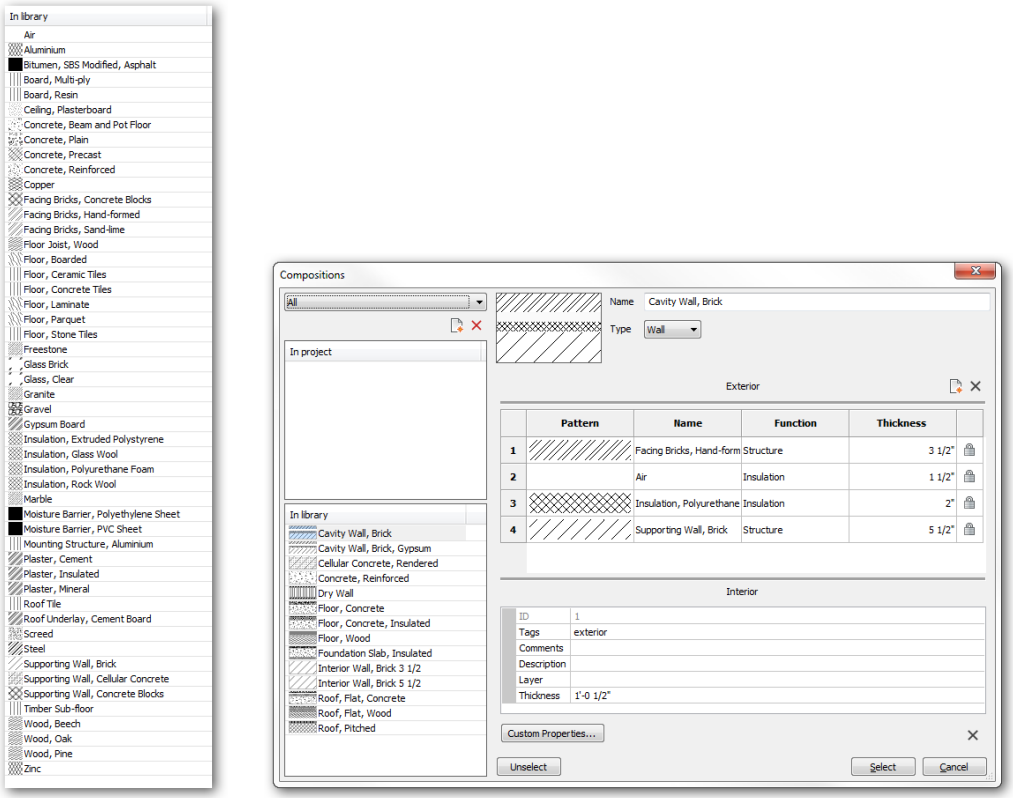
5. Next step is to tell BricsCAD that these are walls. You do this with the **bimClassify** command.
(NEW TO V16) Additional entities were added to classification — wIndow, Door, and Unclassify — and the command's prompt order was changed.


```
: bimclassify
Classify entities as: Wall/Column/Slab/Beam/wIndow/Door/buildingElement/Auto/Unclassify: 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 can be bricks, wood, gyproc (drywall), and so on.

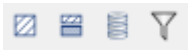
- **Interior** of walls provides strength through 2"x4" (interior walls) or 2"x6" (exterior) studs made of wood or metal. The strength of walls is needed to hold up walls, roofs, and so on. Extra pairs of 2"x6"- or 2"x10"-sized beams, called headers, are needed over window and door openings to distribute 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 above shows the white Tyvek wrap, along with 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 can use one of the dozen wall, floor, and roof compositions provided by BricsCAD, or else define your own with the Compositions dialog box and the 40+ materials provided. (The definition, look, and properties of materials are edited through the Building Materials dialog box.)



***Left:** Pre-defined materials available in BricsCAD; **right:** dialog box for customizing compositions*

TIP The easy way to get to the dialog boxes that define materials and compositions is by clicking their buttons in the BIM Compositions pane:



***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
Enter composition name or [Dialog] <Dialog>: Cavity Wall, Brick, Gypsum
Select entities to attach composition: all
Entities in set: 4
Select entities to attach composition: (Press Enter to finish)
The composition has been assigned to 4 element(s).
```

(Alternatively, you can drag the "Cavity Wall, Brick, Gypsum" composition from the BIM Compositions bar onto each wall, one at a time.)

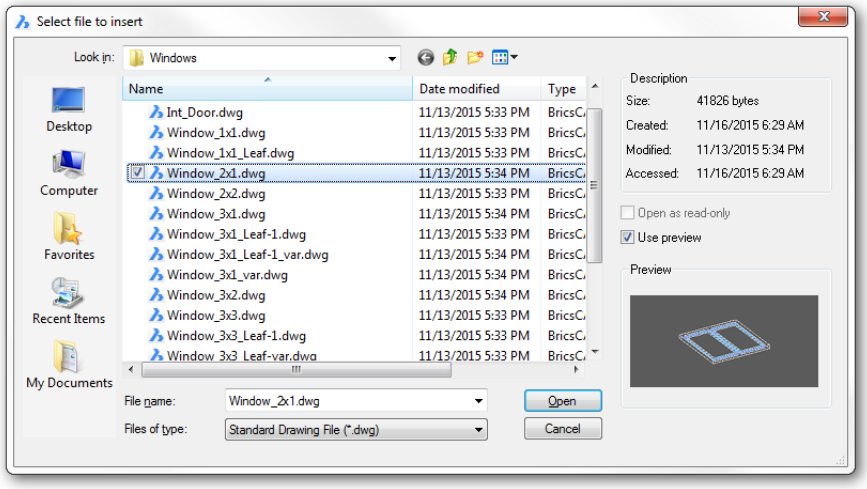
The walls look no different, and changing the visual style doesn't show the bricks either. The BIM Compositions bar, however, lists the composition you applied.

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 small selection of them. Enter the **bimInsert** command:

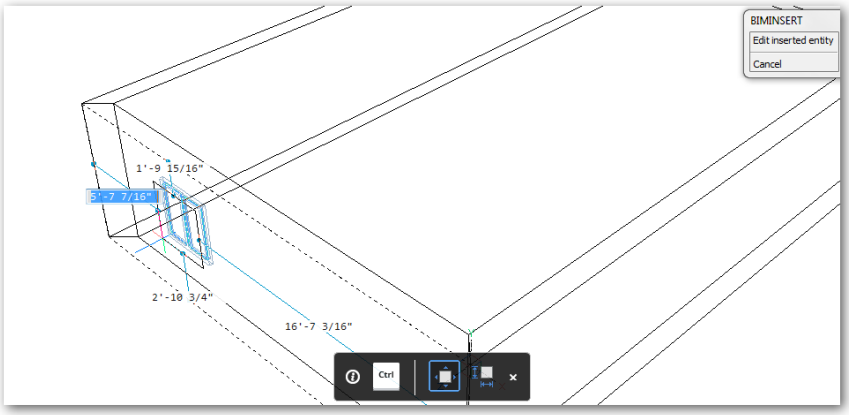

```
: biminsert
```

a. Notice the Select File to Attach dialog box. Choose a window type, and then click **Open**. For this tutorial, I picked the "Window_Fixed_2x1.dwg" file.



Choosing a window type

b. 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:

Insert — dynamically dimensions the location of the window in the wall; press **Tab** to move between the dimension fields; prompts you:

```
Select insertion point or [Edit inserted entity]:
```

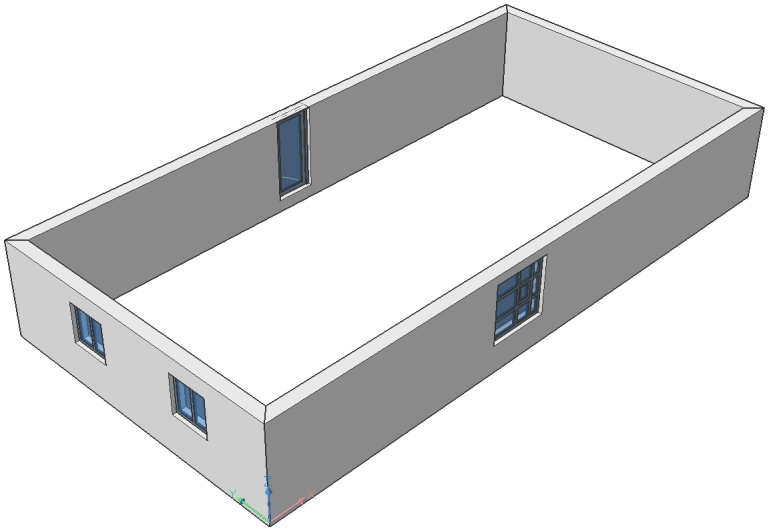
Edit — allows you to change the size of the window:

```
Edit Height [Width/Done]:
```

c. For this tutorial, just insert the window anywhere in the wall:

```
Select insertion point or [Edit inserted entity]: (Click to place the window)
```

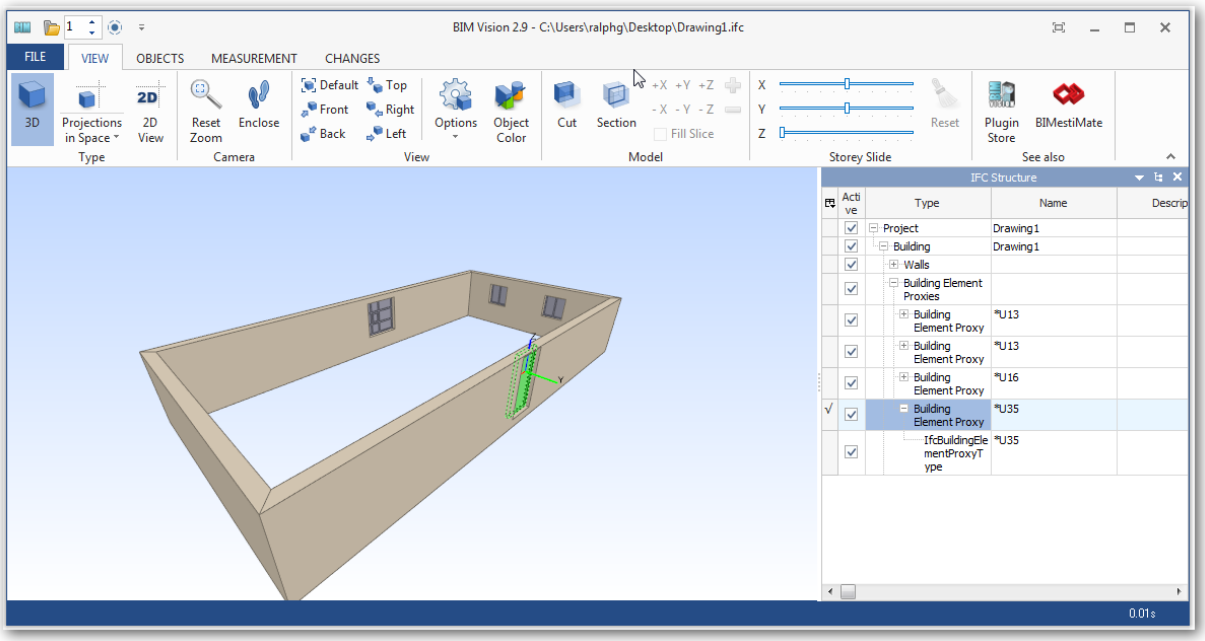
d. Repeat the **bimInsert** command to place more windows and a door. To see a nicer rendering of the building, change the visual style to "BIM."



Placing windows

7. To export the model in IFC format, use the **bimExport** command.

8. To view the resulting IFC file, use an IFC file viewer, such as the free one from http://www.bimvision.eu/home.

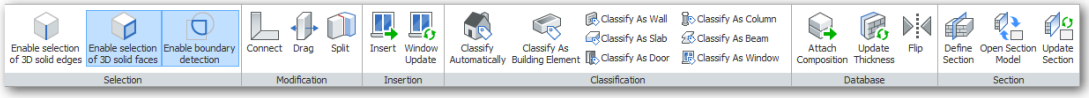


Viewing IFC data with a viewing program

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

Exporting & Importing 3D Models

BricsCAD imports and export the following 3D file formats. From the **File** menu, choose **Import** or **Export**. These exchange formats are included with BricsCAD at no extra cost.

Import	Export	Description
dwg, dxf	dwg, dxf, dwt	AutoCAD drawing, interchange, template
...	dwf	Autodesk Design Web Format
...	pdf	Adobe Page Description Format
dae	dae	Collada Digital Asset Exchange
...	svg	Scalable Vector Graphics
...	stl	Stereolithography
wmf, emf, wmz, emz	...	Windows Meta Format, Enhanced Format

Separately, the **AcisIn** and **AcisOut** commands import and export 3D models as *.sat* files (Save As Text format).

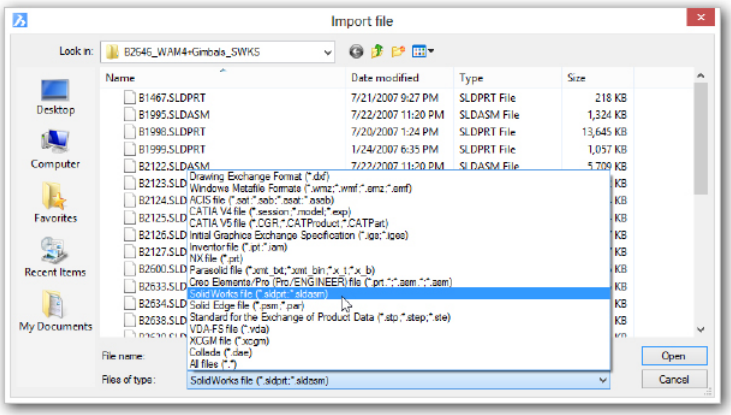
TIP To control the quality of DXF, PDF, and SVG exports, see the **Export** section in the Settings dialog box.

Not listed here are the many raster formats to which BricsCAD and AutoCAD export drawings.

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 license fees that need to be paid for the translators. AutoCAD includes extensive export and import translators at no extra cost.

When Communicator is installed on your computer, the added file format 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	•		Jupiter Technology
x_t, xmt_txt, x_b, xmt_bin	•	•	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
asm, prt	•	•	Creo Elements / Pro Engineer
iam, ipt	•	•	Inventor
prt	•	•	NX
3dm	•		Rhino
par, psm		•	Solid Edge
sldasm, sldprt	•	•	Solidworks

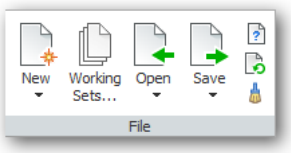
Export Formats Supported

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 Page Description Format
dgn	•		Bentley Systems V7 and V8
model, catpart, catproduct		•	CATIA V4 and V5

(NEW TO V16) When assembly file files are imported, the new **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**

As this chapter illustrates, BricsCAD in many areas of 3D design. is more capable than AutoCAD Bricsys is currently targeting mechanical design, which is why it doesn't offer the industrial design-oriented 3D surfacing commands found in AutoCAD. Expect further 3D capabilities as Bricsys also pursues the AEC-BIM market.

APPENDIX A

Command Name Cross-reference

THIS APPENDIX LISTS THE NAMES OF COMMANDS FOUND IN BRICSCAD V16 AND AUTOCAD 2016. 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 V15 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 found at the end 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		
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	

AutoCAD Command	BricsCAD Command	Notes
Archive	...	
Area	Area	
Array, -Array	Array, - Array	In BricsCAD, now supports dynamic, editable arrays
ArrayClassic	ArrayClassic	
ArrayClose	ArrayClose, -ArrayClose	
ArrayEdit	ArrayEdit	
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

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
...	**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
...	bimReposition	Repositions inserts (doors, windows) in the faces of solids
...	bimSection	Creates sections from BIM models
...	bimSectionOpen	Opens drawing files related to BIM sections

AutoCAD Command	BricsCAD Command	Notes
...	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

Cal	Cal	BricsCAD displays Windows Calculator
Camera	Camera	
...	Center	In AutoCAD, use -Osnap Center
Chamfer	Chamfer	
ChamferEdge	...	In BricsCAD, use dmChamfer

AutoCAD Command	BricsCAD Command	Notes
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
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	
...	CopyEData	In BricsCAD, copies xdata between entities
CopyHist	CopyHist	
CopyLink	...	
CopyToLayer	...	

AutoCAD Command	BricsCAD Command	Notes
...	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

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
DgnAdjust	...	BricsCAD does not import DGN files
DgnAttach	...	
DgnClip	...	
DgnExport	...	

AutoCAD Command	BricsCAD Command	Notes
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
...	**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	Converts watertight region and surface entities to 3D solids
...	**dmTangent3D**	Applies 3D tangency constraints
...	dmThicken	Converts surface to 3D solids with specified thicknesses
...	**dmUpdate**	Updates 3D models to satisfy constraints
...	Dome	In BricsCAD, draws 3D solid domes
Donut	Donut	
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

AutoCAD Command	BricsCAD Command	Notes
...	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	

Dimensions

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	
...	DimStyleSet	Sets the working dimension style
DimTEdit	DimTEdit	

E

EAttEdit	EAttEdit	
EAttExt	...	In BricsCAD, use the DataExtraction command

AutoCAD Command	BricsCAD Command	Notes
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 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	...	In BricsCAD, use the PdfOut command
ExportSettings	...	
-ExportToAutocad	...	
...	ExpUcs	
Extend	Extend	
...	Extension	In AutoCAD, use -OSnap Extension
ExternalReferences / Close	...	In BricsCAD, use the Xref command
Extrude	Extrude	

F

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	
...	Flatten	In BricsCAD, flattens 3D objects with thickness
Freespot	...	In BricsCAD, use the SpotLight command
Freeweb	...	In BricsCAD, use the WebLight command

G

GcCoincident	GcCoincident	Gc = geometric constraint

AutoCAD Command	BricsCAD Command	Notes
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	
...	GradientBkgOn	
GraphicsConfig	...	In BricsCAD, use the RedSdkInfo command
GraphScr	GraphScr	
Grid	Grid	
Group	Group, -Group	
GroupEdit	...	

H

Hatch	Hatch, -Hatch	
HatchEdit	HatchEdit, HatchEdit	
HatchGenerateBoundary	HatchGenerateBoundary	
HatchSetBoundary	...	
HatchSetOrigin	...	
HatchToBack	HatchToBack	
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	...	

AutoCAD Command	BricsCAD Command	Notes
HISettings	...	
Hyperlink	Hyperlink, -Hyperlink	
HyperlinkOptions	HyperlinkOptions	
I		
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		
Join	Join	
JpgOut	...	
JustifyText	...	
L		
LayCur	LayCur	
LayDel	...	In BricsCAD, use Layer command
Layer, LayerPalette, LayerClose	Layer, -Layer	BricsCAD uses Explorer for layers
LayerP	LayerP	
LayerPMode	...	In BricsCAD, use LayerPMode system variable
LayerState	LayerState	BricsCAD uses Explorer for layer states
LayerStateSave	...	
LayFrz	LayFrz	
LayIso	LayIso	
LayLck	LayLck	

AutoCAD Command	BricsCAD Command	Notes
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
...	LicPropertiesCommunication	Transmits license data to Bricsys
...	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

...	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	
MatchCell	...	
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	...	To be supported in a future release of BricsCAD
Materials	Materials	

AutoCAD Command	BricsCAD Command	Notes
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	
MSpace	MSpace	
MtEdit	...	In BricsCAD, use Properties
MText	MText, -MText	
Multiple	Multiple	
MView	MView	
MvSetup	MvSetup	

N

AutoCAD Command	BricsCAD Command	Notes
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

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	
-OSnap	OSnap, -OSnap	
OverKill	OverKill, -OverKill	

P

PageSetup	PageSetup	
Pan	Pan, -Pan	
...	Parallel	In AutoCAD, use -OSnap Parallel
Parameters, ParametersClose	...	In BricsCAD, use bmBrowser

AutoCAD Command	BricsCAD Command	Notes
...	-Parameters	Creates and edits parameters at the command line
PartialLoad	...	
-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	
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	

AutoCAD Command	BricsCAD Command	Notes
PSpace	PSpace	
PType	...	In BricsCAD, use the DdPtype command
Publish	Publish, -Publish	
PublishToWeb	...	
Purge	Purge, -Purge	
Pyramid	Pyramid	
Q		
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		
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	
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

AutoCAD Command	BricsCAD Command	Notes
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	...	BricsCAD does not support creation of dynamic blocks
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

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	
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

AutoCAD Command	BricsCAD Command	Notes
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	
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	

AutoCAD Command	BricsCAD Command	Notes
Style	Style, -Style	BricsCAD uses Explorer for styles
StylesManager	StylesManager	
Subtract	Subtract	
SunProperties / Close	SunProperties	
...	SupportFolder	Opens C:\Users\<login>\AppData\Roaming\Bricsys\BricsCAD\V16x64\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

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\V16x64\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	
TextToFront	TextToFront	
Thicken	...	In BricsCAD, use the DmExtrude command
TifOut	...	
Time	Time	
TInsert	...	
Tolerance	Tolerance	
-Toolbar	Toolbar, -Toolbar	
ToolPalettes / Close	ToolPalettes, ToolPalettesClose	
Torus	Torus	

AutoCAD Command	BricsCAD Command	Notes
TpNavigate	...	
...	Trace	Draws wide lines
Transparency	Transparency	
TreeStat	...	
Trim	Trim	
...	TxtExp	Explodes text

U

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

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**	
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**	

AutoCAD Command	BricsCAD Command	Notes
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

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
WsSave	WsSave	
WsSettings	WsSettings	
...	WvTile	In BricsCAD, tiles windows vertically

X

XAttach	XAttach	
XBind	...	To be supported in a future release of BricsCAD
XClip	XClip	
XEdges	...	

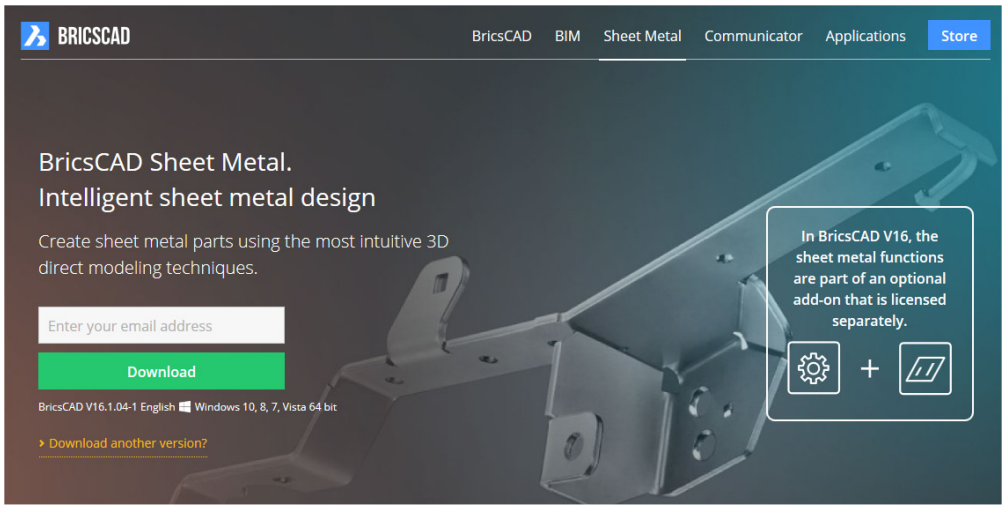
AutoCAD Command	BricsCAD Command	Notes
XLine	XLine	
XOpen	XOpen	
Xplode	Xplode	
Xref, -XRef	XRef, -XRef	Explorer for external references
Z		
Zoom	Zoom	
#		
...	2dIntersection	In AutoCAD, use -OSnap Intersection
3D	3D	
3dAlign	...	
3dArray	3dArray	
3dClip	...	
3dCOrbit	...	In BricsCAD, use the RtRot command
...	3dConvert	In BricsCAD, 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
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 V16. See https://www.bricsys.com/en_INTL/sheetmetal.

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
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
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
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



Bricsys Web page for obtaining the Sheet Metal add-on

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				
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
...		*(not used)*	AxisMode	Toggles axis display
...		*(not used)*	AxisUnit	Specifies axis units

B

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		...	
BlockEditor	0		...	
...		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		...	

C

...		1	CacheLayout	Toggles caching of layouts
CacheMaxFiles	256		...	
CacheMaxTotalFiles	1024		...	
CalcInput	1		...	
CameraDisplay	0	0	CameraDisplay	
CameraHeight	0	0	CameraHeight	
CAnnoScale	1:1	1:1	CAnnoScale	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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
...		https://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
...		https://www.mychapoo.com	**ChapooWebsite**	Names the Chapoo Web site
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
...		#ffffdd	**CmdLineListBgColor**	Specifies command line background color
...		#000000	**CmdLineListFgColor**	Specifies command line foreground color

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		:	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	

D

DataLinkNotify	2		...	
Date	2455035.85	2455035.63	Date	
DbcState	0	0	DbcState	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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
DefIPlStyle	bycolor	ByColor	DefIplstyle	
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		...	
DivMeshConeAxis	8		...	
DivMeshConeBase	3		...	
DivMeshConeHeight	3		...	
DivMeshCylAxis	8		...	
DivMeshCylBase	3		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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	
DynConstraintDisplay	1		...	
DynConstraintMode	1	1	DynConstraintMode	
DynDiGrip	31	31	DynDiGrip	
...		142	**DynDimColorHot**	Specifies dynamic dimension hot color
...		142	**DynDimColorHover**	Specifies dynamic dimension hover color

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		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		...	

Dimensions

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	
DimExe	0.18	0.18	DimExe	
DimExo	0.06	0.06	DimExo	
DimFit	3	3	DimFit	
DimFrac	0	0	DimFrac	
DimFxl	1	1	DimFxl	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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	
DimTxRuler	on		...	
DimTxSty	standard	STANDARD	DimTxSty	
DimTxt	0.18	0.18	DimTxt	
DimTxtDirection	off		...	
DimTzin	0	0	DimTZin	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
DimUnit	2	2	DimUnit	
DimUpt	off	0	DimUpt	
DimZin	0	0	DimZin	
E				
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				
FacetErDevNormal	40		...	
FacetErDevSurface	0		...	
FacetErGridRatio	0		...	
FacetErMaxEdgeLength	0		...	
FacetErMaxGrid	4096		...	
FacetErMeshType	0		...	
FacetErMinUGrid	0		...	
FacetErMinVGrid	0		...	
FacetErPrimitiveMode	1		...	
FacetErSmoothlev	1		...	
FacetRatio	0	0	FacetRatio	
FacetRes	0.5	0.5	FacetRes	
...		1	FeatureColors	Colors solid faces by related features
FbxImportLog	1		...	
FieldDisplay	1	1	FieldDisplay	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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

GalleryView	1		...	
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	
...		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	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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 wireframes
...		0	GsDeviceType3D	Specifies graphics system for hidden, etc.
GtAuto	1		...	
GtDefault	0		...	
GtLocation	1		...	
H				
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	
HpBackgroundColor	"."		...	
HpBound	1	1	HpBound	
HpBoundRetain	0		...	
HpColor	"."		...	
HpDlgMode	2		...	
HpDouble	0	0	HpDouble	
HpDrawOrder	3	3	HpDraworder	
HpGapTol	0	0	HpGapTol	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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

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	
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	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
IntersectionDisplay	off	0	IntersectionDisplay	
ISaveBak	1	1	ISaveBak	
ISavePercent	50	50	ISavePercent	
Isolines	4	4	Isolines	
L				
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		...	
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	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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

...		0	MacroRec	Reports that macro is being recorded
MacroTrace	0	0	MacroTrace	
...		0.01	**MassPropAccuracy**	Specifies accuracy for mass properties
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	
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

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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

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

ObjectIsolationMode	0	0	ObjectIsolationMode	
ObscuredColor	257	257	ObscuredColor	
ObscuredLtype	0	0	ObscuredLtype	
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		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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

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	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
...		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	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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		...	
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	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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
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	

Q

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

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		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

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
...		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**	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		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
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

SafeMode	0		...	
...		1	**SaveChangeToLayout**	Saves print changes to layout

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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
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		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
...		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		...	
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

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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	
SubObjSelectionMode	0		...	
SunStatus	0		...	
SurfaceAssociativity	1		...	
SurfaceAssociativityDrag	1		...	
SurfaceAutoTrim	0		...	
SurfaceModelingMode	0		...	
SurfTab1	6	6	Surftab1	
SurfTab2	6	6	Surftab2	
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	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
SysMon	1		...	

T

...		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	
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	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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

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	
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		...	
User1-5	0	0	User1-5	
UserR1-5	0	0	UserR1-5	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
UserS1-5	""	""	UserS1-5	
...		2	UseSheetMetal	Determines the sheet metal license type
...		0	**UseStandardOpenFileDialog**	Displays additional folder in file dialog boxes

V

...		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		...	
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)"		...	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
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

...		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
...		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	

AutoCAD System Variable Names	AutoCAD's Default Values	BricsCAD's Default Values	BricsCAD Preference & System Variable Names	Notes on Variables Unique to BricsCAD
WsCurrent	2D drafting & annotation	2D Drafting	WsCurrent	
X				
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		...	
XRefType	0		...	
Z				
ZoomFactor	60	60	ZoomFactor	
ZoomWheel	0	60	ZoomWheel	
#				
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 V16. 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\V16x64\en_US\Support*

Mac in folder */Users/<login>/Library/Preferences/Bricsys/BricsCAD/V16x64/en_US/Support*

Linux in folder *home/<login>/Bricsys/BricsCAD/V16x64/en_US/support*

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
A			
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			
...		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			
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
Dimensions			
...		dimension	Dim

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
DimAligned	dal, dimali	dal, dimali	DimAligned
DimAngular	dan, dimang	dan, dimang	DimAngular
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

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

Fillet	f	f	Fillet
Filter	fi		...
FlatShot	fshot		...

AutoCAD Command	AutoCAD Alias	BricsCAD Alias	BricsCAD Command
G			
GeographicLocation	geo, north, northdir	geo	GeographicLocation
GeomConstraint	gcon		...
Gradient	gd		...
...		g	Grid
-Group	-g		...
Group	g		...
H			
-Hatch	-h	-h, -bh	-Hatch
Hatch	h, bh	h, bh	Hatch
HatchEdit	he	he	HatchEdit
HatchToBack	hb		...
Hide	hi	hi	Hide
HidePalettes	poff		...
I			
...		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			
Join	j		...
L			
-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

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
Lengthen	len	len, editlen [ICAD]	Lengthen
...		lighting	Light
...		ll	LightList
Line	l	l, 3dline	Line
-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

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

NavSMotion	motion		...
NavSMotionClose	motioncls		...
NavSWheel	wheel		...

AutoCAD Command	AutoCAD Alias	BricsCAD Alias	BricsCAD Command
NavVCube	cube	navvcube	LookFrom
NewShot	nshot		...
NewView	nview		...
...		ddnew	NewWiz

O

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

-Pan	-p		...
Pan	p	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

QLeader	le		...
...		n	QNew
...		qt	QText

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
QuickCalc	qc		...
QuickCui	qcui		...
Quit	exit	exit	Quit
QvDrawing	qvd		...
QvDrawingClose	qvdc		...
QvLayout	qvl		...
QvLayoutClose	qvlc		

R

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

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		

AutoCAD Command	AutoCAD Alias	BricsCAD Alias	BricsCAD Command
SheetSet	ssm	ssm	SheetSet
ShowPalettes	pon		...
...		freehand ICAD	Sketch
Slice	sl	sl	Slice
Snap	sn	sn	Snap
Solid	so	so, plane ICAD	Solid
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

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

Ucs	dducs		...
UcsMan	uc, dducs, dducsp		...
Union	uni	uni	Union
UnisolateObjects	unhide, unisolate	unhideobjects, unhide, unisolate	UnisolateObjects

AutoCAD Command	AutoCAD Alias(es)	BricsCAD Alias(es)	BricsCAD Command
-Units	-un	-un	-Units
Units	un, ddunits	un, ddunits	Units
V			
...		vba	VbaIde
-View	-v	-v	-View
View	v, ddview	v, ddview, expviews [ICAD]	View
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			
-WBlock	-w		...
WBlock	w, acadwblockdialog	w	WBlock
...		closeall	WCloseAll
Wedge	we	we	Wedge
...		wi	WmfIn
...		wo	WmfOut
X			
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			
Zoom	z	z	Zoom
3			
3dAlign	3al		...
3dArray	3a	3a, array3d	3dArray
3dFace	3f,	3f, face	3dFace
...		mesh	3dMesh
3dMove	3m		...

AutoCAD Command	AutoCAD Alias	BricsCAD Alias	BricsCAD Command
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 V16. To learn how to customize all aspects of BricsCAD, see the *Customizing BricsCAD* ebook available for purchase from the www.Bricsys.com 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.

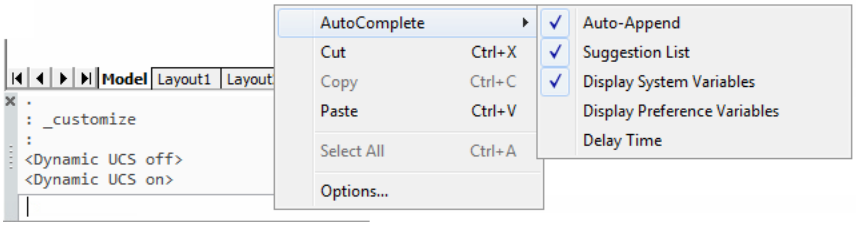
AutoCAD Action	AutoCAD Command	Shortcut Keystrokes	BricsCAD Command	BricsCAD Action
Erases selected objects	Erase	Del	Erase	Erases selected objects
	...	PageUp	Pan PgU	Pans up
	...	PageDown	Pan PgD	Pans down
	...	Shift+Left	Pan PgL	Pans left
	...	Shift+Right	Pan PgR	Pans right
	...	Shift+Up	Pan PgU	Pans 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.

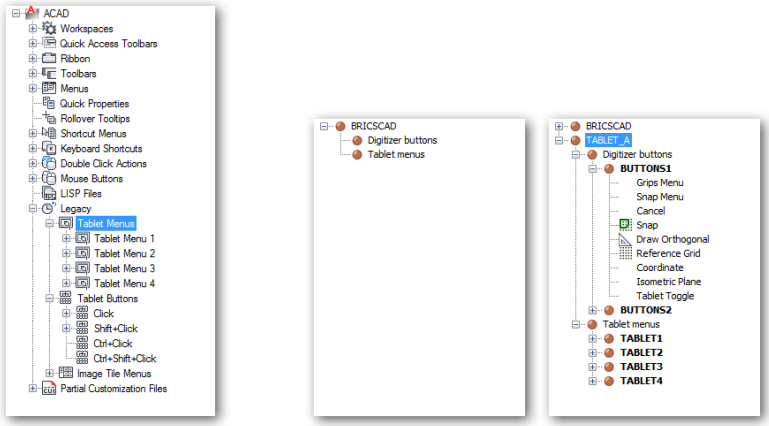
TABLET BUTTONS

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 www.bricsys.com/bricscad/download//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.

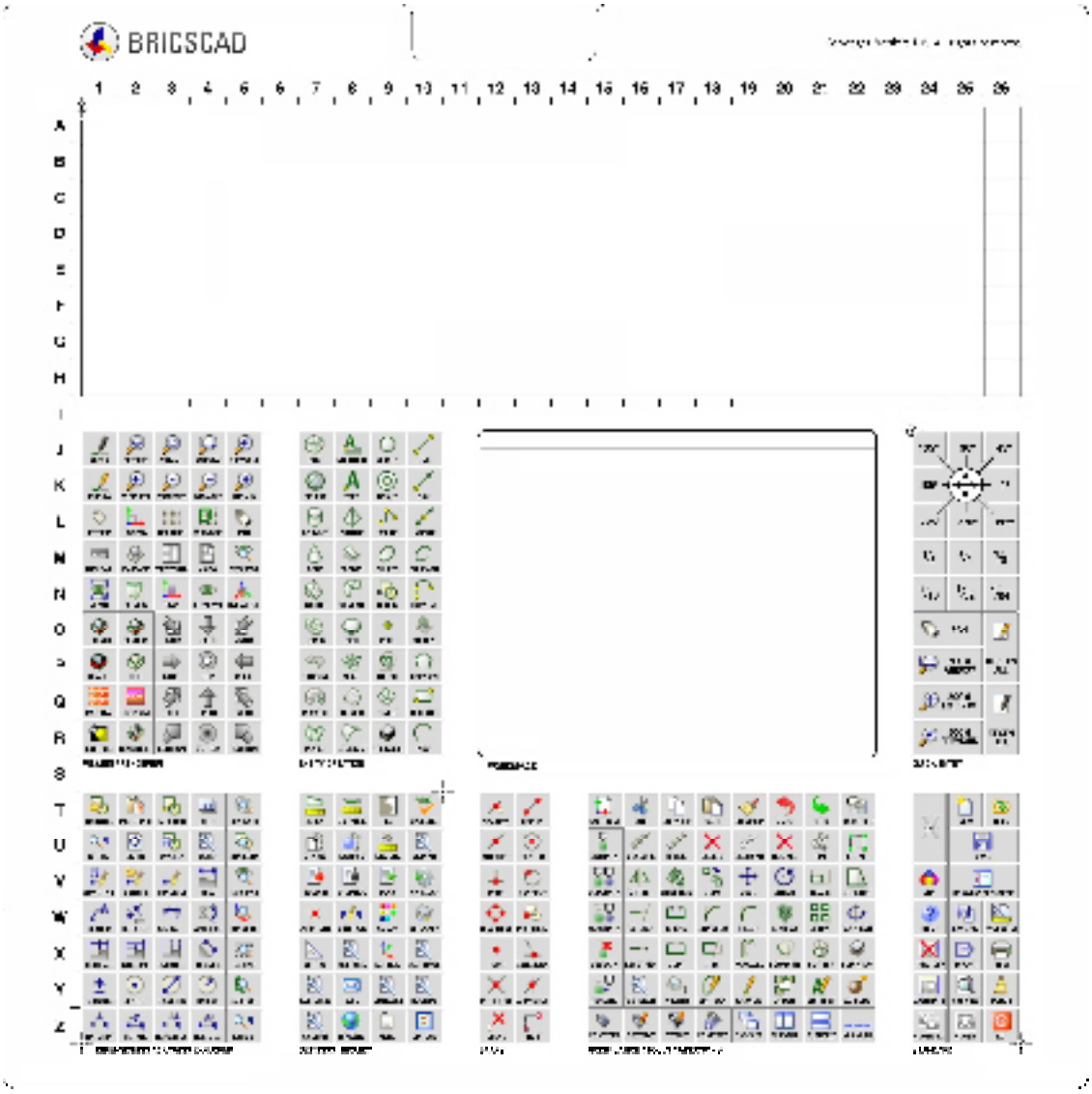


Left: *Tablet button definitions in AutoCAD's CUI dialog box.*
Center: *Default tablet definition in BricsCAD's Customize dialog box.*
Right: *Tablet definition in BricsCAD after loading "tablet(acadLike).cui."*

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.

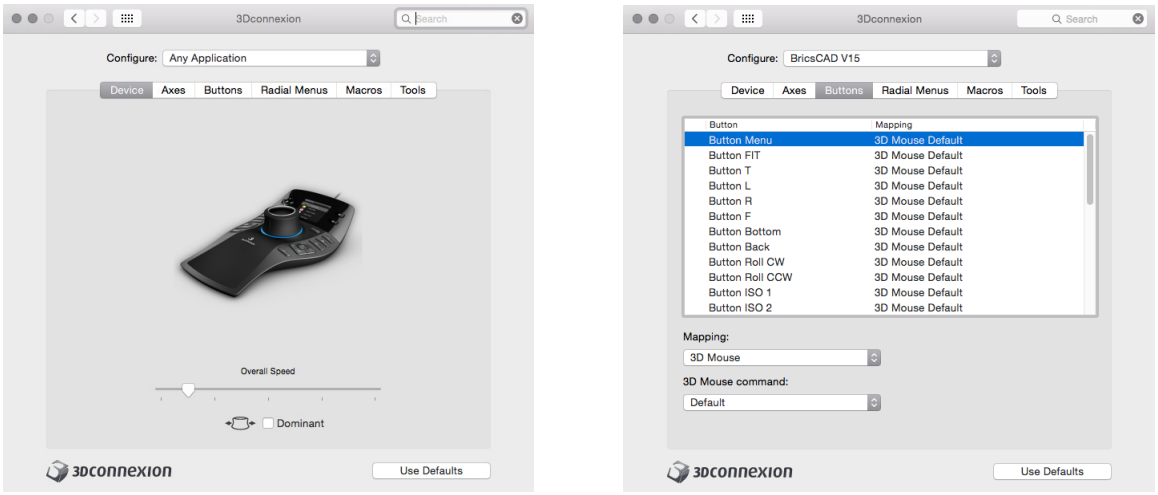
Function	Windows & Linux Button and Key	Mac Button and Key	BricsCAD Command or System Variable Executed
Moves forward, backwards, left, or right	Alt + Left button	Opt + Left button	RtWalk
Moves up, down, or sideways	Alt + Middle button	Opt + Middle button	RtUpDown
Looks around	Ctrl + Middle button	Cmd + 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 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.