WindowIn on DataCAD

The Newsletter for DataCAD Users

Volume 4, Number 10 | *Price $8.00* | *May1991*

Editor's Notes

The Last Issue of
"WindowIn on DataCAD"

Since our announcement for terminating publication of "WindowIn on DataCAD" many of you have taken the time to call and/or write us. We thank you for the appreciation which you have expressed. In my experience, the DataCAD community is a unique family. Again, we have enjoyed the many acquaintances and friendships that have evolved over the years.

The refunds for the remaining issues on your subscriptions have been mailed under separate cover. Along with those refunds, we have included a letter which describes how your refund was calculated. Should we have erred, please contact us at your earliest convenience.

A/E/C Systems '91 in Washington, D.C.

In summary, the attendance at "A/E/C Systems '91" was the lowest that I have seen over the past several years. Perhaps, it is a reflection on the state of the economy.

Other than the extent of the AutoCAD presence, a phenomenal graphics card by Vermont Microsystems, Inc. and increased sophistication in the scanning industry, the show lacked the luster that I have associated with it in the past.

With respect to DataCAD, CADKEY unveiled the objective of their long-range development. It is a "Windows" based product which they state will be available in early 1992. Phil Hart, in his article on page 10, provides the available details on this product. In addition, he has provided coverage of the National DataCAD User Group Meeting.

This Issue

Due to the time/space required to cover "Dimensioning Procedures", we were unable to complete it for this issue. Hopefully, our previous coverage of dimensioning options will provide insight into proper procedures.

A GLIMPSE INSIDE

Section — *Page*

Dr. DataCAD
By Bill D'Amico

DataCAD Version 4.06 — 3

THE "ViewMast" MACRO: Part III
By The Staff at WindowIn on DataCAD

Introduction — 4

THE "Fly Thru" FUNCTION
By The Staff at WindowIn on DataCAD

Introduction — 4
Defining "Fly Thru" Views — 4
Removing/Resetting "Fly Thru" Views — 5
Graphic Representation of "Fly Thru" Views — 6
Displaying The "Fly Thru" Views — 6
Creating 3D Views from "Fly Thru" Views — 6

THE "Fly Along" FUNCTION
By The Staff at WindowIn on DataCAD

Introduction — 7
Drawing the Contour to Establish The Path of the Viewer's Eye — 7
Options for Defining Views in "FlyAlong" — 8
Converting Temporary Views to 3D Views — 9
Example Showing Eye Path with Varying Z-Coordinates — 9
Warning — 9

WindowIn/Extents
By Phil Hart

A/E/C Systems '91 — 10
The Parthenon Project — 10
Features Included in Parthenon Project — 10
Other News from the User's Meeting — 11
At the Booth — 11
The Vintage Group Products — 12
cadKEYNOTE — 13
conTEXT — 13
Third Party Products at A/E/C — 14
In Closing — 14

New User Group

Upstate New York — 14

New User Group

Information on a new user group being formed for upstate New York is included on page 14.

PUBLICATION INFORMATION

SUBSCRIPTIONS

Annual subscriptions to **WindowIn on DataCAD** are available for $72.00 if within U. S. A., $85.00 in Canada and $106.50 elsewhere. You will receive Twelve (12) issues plus the **Annual Index**. To subscribe, simply list contact name, company name, address and telephone number. Mail with payment to **WindowIn on DataCAD**, P. O. Box 502, Middlebury, VT 05753.

BACK ISSUES

Back issues of **WindowIn on DataCAD** may be purchased at a cost of $8.00 each ($9.00 in Canada and $10.00 elsewhere). Complete sets for Volume 1 OR Volume 2 are available for $60.00 ($66.00 in Canada, $80.00 elsewhere).

PAYMENT

ALL PRICES QUOTED ARE IN U. S. FUNDS. Checks for all materials should be made payable in U. S. funds to: WindowIn on DataCAD PLEASE NOTE: ALL ORDERS OUTSIDE OF THE U. S. A. AND ORDERS FOR BACK ISSUES MUST BE PREPAID.

VOLUME 1: (August '87 - July '88)

No. 1 - 8 pp. (Use of REFERENCE POINT)
No. 2 - 16 pp. (WALLS and related commands)
No. 3 - 16 pp. (STRETCH and DEFAULT DRAWING)
No. 4 - 16 pp. (Scope of Drawings and Use of Layers)
No. 5 - 12 pp. (GOTOVIEW Menu and Use of Views)
No. 6 - 16 pp. (Default SCALES, DISTANCES and ANGLES)
No. 7 - 16 pp. (Plotting: Part I)
No. 8 - 14 pp. (Plotting: Part II)
No. 9 - 16 pp. (Plotting: Part III)
No. 10 - 16 pp. (RAM above 640KB & DOS Directory Structure)
No. 11 - 16 pp. (Large Drawings, Keyboard Interrupts/Macros)
No. 12 - 16 pp. (Disk Fragmentation and Display menu)

VOLUME 2: (August '88 - July '89)

No. 1 - 16 pp. (Temp./Symbols Part I, DOS CHKDSK)
No. 2 - 14 pp. (Temp./Symbols Part II, Creating Templates)
No. 3 - 14 pp. (Temp./Symbols Part III, Saving Symbols)
No. 4 - 16 pp. (Temp./Symbols Part IV, Placing/Editing Symbols)
No. 5 - 16 pp. (Temp./Symbols Part V, Using Report Forms, Editing Symbols)
No. 6 - 16 pp. (Using the Identify Command)
No. 7 - 16 pp. (Using and Creating "Line Types")
No. 8 - 16 pp. (DataCAD, Microtecture & Sigma Design)
No. 9 - 16 pp. (Keyboard, Mouse/Object Snap, Reference Pt. & Keyboard Entry)
No. 10 - 16 pp. (Implementing DataCAD, Default Drawings)
No. 11 - 16 pp. (Facility Notebook, 3D Viewer)
No. 12 - 16 pp. (System Admin. Procedures, 3D Slabs)

VOLUME 3: (August '89 - July '90)

No. 1 - 16 pp. (Constructing Archs in 3D, System Admin Part II)
No. 2 - 16 pp. (Intro. to DXF; 3D Arcs, Slabs, Fast3D; & Reprint)
No. 3 - 16 pp. (Entity Properties/DXF; Roofs & Inclined Polygons)
No. 4 - 16 pp. (2D Entities/DXF; Inclined Polygons; Rendering with Velocity)
No. 5 - 16 pp. (Roofs with Overhang; Table for distances by pitch)
No. 6 - 16 pp. (DXF: 3D Entities; 3RD Party Products; Hidden Line Removal)
No. 7 - 16 pp. (DXF: Dimen/Symbols; Polylines/Rev Surfs; Review: Keynote; DCAL)
No. 8 - 22 pp. (Hatching; Contours; Potential with DCAL; Review: Blocker)
No. 9 - 20 pp. (3D Models Strategy; DCAL & Entities; Northgate 386/33; Contours)
No. 10 - 20 pp. (Hatching: Part II; DCAL: 3D Entities; Graphics Cards, Review: KB Performance & Command Performance)
No. 11 - 20 pp. (Hatching: Part III; DCAL: Symbols; Creating 3D Text, AEC Systems Show, Dr. DataCAD)
No. 12 - 16 pp. (Hatching: Part IV; DCAL: Template; Digitizing without Digitizer; Tips & Techniques, Dr. DataCAD)

VOLUME 4: (August '90 - Current)

No. 1 - 16 pp. (Installing/Configuring DataCAD Version 4.0, DOS Memory Usage)
No. 2 - 18 pp. (Review: Roofer vs. Roof Builder; DCAL 4.0; QEMM-386)
No. 3 - 16 pp. (The Software Display List)
No. 4 - 16 pp. (DCAL: Attributes; Keyboard Interrupts;Virtual Disks))
No. 5 - 16 pp. (DCAL: Mode_Type; Review: Set_Text; Dr. DataCAD; CAD Environment)
No. 6 - 20 pp (Dimensions Pt 1; 3D GoToViews; Default Drawings)
No. 7 - 16 pp (DCAL; Review: Organic CADD; Viewmast; Dr. DataCAD, DOS Environ.)
No. 8 - 16pp (Dimensioning Part II, including dimension styles, Review: Set-Lines)
No. 9 - 14 pp. (ViewMast Part II, Third Party Products)

LOOKING AHEAD TO JUNE

- This is the last issue of WindowIn on DataCAD to be published.

Topics of Future Issues

None

PUBLICATION STAFF

Editor
Christian L. Davis

Subscriptions and Production
Sue Russ

Contributing Authors
Bill D'Amico
Casco Systems
Freeport, ME
Phil Hart
Moore Weinrich Architects
Brunswick, ME

Editorial Advisory Board
Gutting Horton Architects
Burlington, VT
Peter Gutting
Fritz Horton
Bill Gardner

Doctor DataCAD

By Bill D'Amico (Casco Systems, Freeport, ME)

Greetings-and farewell. This is our first visit in a few months, and I guess it will be our last. I have to admit to a bit of writers block these last 90 days or so, as the doldrums of winter have combined with my learning of the impending demise of "*WindowIn on DataCAD*" to fill my head with thoughts of many things besides any current bugs in DataCAD. We'll discuss those thoughts in a few minutes...

As for the bugs, I must say that aside from the pet peeves I discussed in our last meeting, DataCAD 4.06 has proved to be a very stable release. I don't believe that I would be going out on a limb to say that if you have been holding off on installing 4.06, you shouldn't hold off any longer. It fixes several problems with the **Bearing** angle type for those who are doing site work or using any angle type besides **Normal**. Also, it has eliminated a number of problems reported here in "*Doctor DataCAD*":

(1) The **Copy** command would get stuck in an infinite loop under certain cases if **LyrSrch** was on.

(2) Some drawings created in DataCAD v3.6 would generate a "*Fatal Error 2136*" when loaded into v4.00.

(3) Some drawings created in DataCAD v3.6 would generate a "*Fatal Error 2100*" when loaded into v4.00 and **Enlarge/NewCenter/Invert** was selected.

If you are still running DataCAD 4.0 and are experiencing no problems whatsoever, you will probably not reap any great benefits from upgrading to 4.06, so you should do so at your leisure. However, *ANYONE* using DataCAD 4.05 should upgrade to 4.06 ASAP! This is probably a very late warning to most people, as if you have received 4.05 from CADKEY and installed it, you probably very quickly ran into the very nasty crashes that would occur when working with associative dimensions. *DO NOT USE DATACAD 4.05 UNDER ANY CIRCUMSTANCES!* I would recommend discarding (or reformatting and reusing) any and all 4.05 disks sent to you by CADKEY.

If and when you are considering upgrading to 4.06, you might want to note that the "*REVHIST.DOC*" file (the file on disk 1 that documents changes between versions of DataCAD) contains reference to the 22 "*CHANGES FROM DATACAD 4.00 TO 4.05*". This is a typo. It should read "*CHANGES FROM DATACAD 4.00 TO 4.06*".

Now, for my thoughts about the end of "*WindowIn on DataCAD*".

Chris Davis - the editor of "*WindowIn*" - has given an enormous amount of time and energy to the DataCAD community. His efforts long ago went above and beyond a business and into the realm of a labor of love. Chris has done everything he could to provide timely and greatly needed technical information about the DataCAD product that you could find nowhere else.

All of us who have written for "*WindowIn*" have given what we could to a devoted, receptive base of users that for years have stood by their product of choice in the face of the "*big guns*" of the CAD market. Chris, however, has easily given more than the rest of us combined. DataCAD is still a wonderfully conceived product, but it has lost a valuable support vehicle with this last issue of "*WindowIn*".

I suppose that if I were to have a parting statement to the DataCAD user base, it would be to experience the software you use to the best of your ability. Use whatever software you now own - and will own - with an entirely open mind. Don't become a slave to it, and don't suffer in silence. Seek help for the problems you encounter, as you are likely not alone, and always look for new ways, and for new software, to help you get your work done.

And, for many of you who read this, this will not be the end of our communications. I am grateful that I have made many friends and contacts through this column and through my DCAL macro work that I intend to continue. And, for those with whom I have not had contact, please do not hesitate to call with DataCAD related issues and problems. I can be reached at the following address:

Bill D'Amico
Casco Computer Systems, Inc.
317 US Route 1 South
Freeport, ME 04032
(207) 865-4399

Best wishes to all...

"ViewMast" MACRO: PART III

The Staff at WindowIn on DataCAD

INTRODUCTION

This issue of *WindowIn on DataCAD* culminates the series on the *"ViewMast"* macro supplied with DataCAD. Previous issues have discussed the commands in the macro which are equivalent to those found on the **GoToView** menu (Vol. 4, No. 7) and the **AutoView, AutoHide** and **Slides** functions (Vol. 4, No. 9).

This article will complete our review of this macro by discussing the **Fly Thru** and **FlyAlong** functions. In summary, these functions provide different methods for defining a sequence of *perspective* views. These views can then be added to the list of *"3D Views"* which are managed via the commands under the 3D version of **GoToView**.

Once the views have been defined, one can *preview* them using the commands in the **Fly Thru** and **FlyAlong** functions. With these commands, however, the *"lines"* which should be hidden will be displayed. In other words, a *"hidden line removal"* is not actually performed for the views defined.

As a result, the real power behind these functions lies in the ability to define a sequence of *perspective* views. Previewing them ensures that a proper and meaningful sequence has been obtained. Once the views are saved as *"3D Views"*, the order and sequence can be further refined using the commands on the 3D version of the **GoToView** menu, or their equivalent in the *"ViewMast"* macro.

The resulting *"3D Views"* can then be processed via the **AutoHide** function in order to automate the process of removing hidden lines. By selecting *"layer files"* in the **AutoHide** function as the destination for the results of the hidden line removal, one can then develop a *"slide show"* using the **Slides** function. Through this process, the *"slide show"* will depict a *"fairly"* realistic sequence of views with hidden lines removed.

In summary, the *"ViewMast"* macro provides an integrated set of functions for:

- Defining 3D Views,
- Developing an appropriate sequence of 3D Views,
- Automating the process of removing hidden lines from a series of views, and
- Developing a *"slide show"* for displaying the views on the monitor.

With the above as an incentive, let's now take a look at the **Fly Thru** and **FlyAlong** functions.

THE *"FLY THRU"* FUNCTION

INTRODUCTION

The **Fly Thru** function of the *"ViewMast"* macro permits one to define one or more *perspective* views by identifying the point of view, or *"eye point"* and the *center of the view*. The resulting *"views"* will be referred to as *"temporary views"* to distinguish them from the views created under the **GoToView** menu.

Once temporary views have been defined, they can be added to the *"3D Views"* maintained via the commands on the **GoToView** menu or their equivalents in the *"ViewMast"* macro. This permits the temporary views to be processed by the commands in the **AutoHide** function of the *"ViewMast"* macro.

DEFINING "FLY THRU" VIEWS

The **Fly Thru** function permits one to define temporary views which are oriented around either a fixed or a varying center point of view. Temporary views are defined with either of the following commands:

DefFixCt: Orients the temporary views around a *"fixed"* point for the *center of the view*. The *eye point*, however, is identified for each individual temporary view.

DefSqunc: Permits one to specify a unique *center of view* and *eye point* for each temporary view.

Once either **DefFixCt** or **DefSqunc** has been selected, several options will be presented in the **Command Win-**

dow for defining each temporary view. These options are described below:

Center: This option will be available *only* under the **DefFixCt** command. The **Center** option enables one to define a "*fixed center point*" for subsequent temporary views to be defined. Altering the "*center point*" will not affect the temporary views already defined. The **Center** command is not available for the **DefSqunc** function, since each temporary view can have a unique *center of view*.

Center Z: Enables one to establish the "*Z height*" for the *center of the view*. The value entered for **Center Z** will be used for subsequent temporary views defined. It is important to note that one can change the value for **Center Z** for different views. This will give the appearance of the viewer looking up and/or down from the same "*eye point*" or from different "*eye points*".

EyePnt Z: Enables one to establish the "*Z height*" for the *eye point*. The value entered for **EyePnt Z** will be used for any subsequent temporary views defined. As with **Center Z**, one can change the value for **EyePnt Z** for subsequent temporary views. For example, one could alter the location and **EyePnt Z** to simulate walking up a flight of stairs.

ConeAng: Enables one to establish the "*angle of the viewing cone*". As illustrated in Figure 1, the angle entered restricts the amount of the model which can be viewed from the *eye point*. Specifically, with larger angles, more of the model will be visible. The value entered for **ConeAng** will be used for any subsequent temporary views defined. As with other options, one can change the value for **ConeAng** as temporary views are defined.

Once appropriate values for the above options have been defined, one can proceed with identifying the "*eye point*" and "*center point*", if necessary, for the next temporary view. If the **DefFixCt** command is being used, one will be prompted to "*Enter eye point of next fly thru view.*". If the **DefSqunc** command is being used, one will be prompted to "*Enter eye point of next fly thru view.*" **and** to "*Enter center point of next fly thru view.*".

Temporary views can be defined by either *or both* **DefFixCt** and **DefSqunc**. With the exception of **Center**, all options set for one command will remain in effect for the other. Further, one can alter any of the available options as temporary views are defined. The new values will be in effect for subsequent views; but will not affect any temporary views already defined.

REMOVING/RESETTING "FLY THRU" VIEWS

If an error is made in defining a temporary view with either **DefFixCt** or **DefSqunc**, one can remove that view by selecting **Backup** from the **Command Window**. It is important to note that **Backup** will always remove the *last* view. Selecting **Backup** several times in sequence has the effect of removing the same number of temporary views from the end of the list of those defined. For example, if five views have been defined and **Backup** is selected three times, the last three views defined will be removed.

When **Backup** is selected, the "*graphic representation*" of the last view defined will be removed from the **Drawing Window**.

Selecting **Reset** from **DefFixCt**, **DefSqunc** or the main menu of **Fly Thru** removes all temporary views defined previously. When **Reset** is selected, the "*graphic representations*" for the views defined *will not*, however, be removed from the **Drawing Window**. To clear them, you must press the "*Esc*" key to refresh the **Drawing Window**.

The **Backup** and **Reset** commands are the only ones available for editing temporary views. For example, assume one defines ten temporary views; and, then, determines that the third view is not desirable. There is no method within the **Fly Thru** function for removing this view. In such cases, one could convert the temporary views to "*3D Views*" using the **AddViews** command. Then, the commands available under **GoToView** could be used to "*rearrange*" or "*delete*" the "*3D Views*".

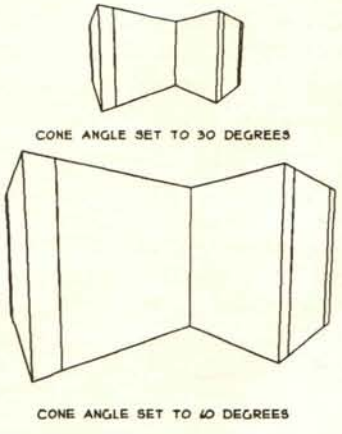


Figure 1
Effect of ConeAng on Generating Temporary Views

GRAPHIC REPRESENTATION OF "FLY THRU" VIEWS

As each temporary view is defined, DataCAD will draw what appears to be an arrow. The appearance of the "*arrow*" is confusing since it "*points*" in the direction opposite from what one would normally expect. Specifically, the "*tail*" is at the *center point* and the "*head*" is at the *eye point*. This is illustrated in Figure 2 for two temporary views defined using the **DefFixCt** function.

The DataCAD representation of each temporary view *should NOT be looked upon as an arrow depicting the direction of the view.* Instead, the "*cone of vision*" is shown at the "*eye point*" with a line connecting the "*eye point*" and the "*center point of the view*". Figure 2 illustrates this graphic representation.

The graphic representations of the temporary views displayed by DataCAD are simply drawn on the screen. Any DataCAD commands which result in the screen being refreshed (e.g. "*Esc*", "*PgUp*", "*PgDn*", etc.) or the display list updated (the letter "*U*") will result in the graphic representations being cleared from the screen. The actual definitions of the views, however, will not be affected. The graphic representations for the views currently defined can be redrawn by selecting the command **DispVwes** from the **Fly Thru** menu.

DISPLAYING THE "FLY THRU" VIEWS

After one or more temporary views have been defined, one can "*preview*" their sequence by selecting **RunViews**. The temporary views will be displayed in the sequence in which they were defined and in accordance with the options selected for each. One can adjust the time that a given temporary view will be displayed by selecting **Delay**. Then, in response to the prompt, enter the number of seconds that each view is to be displayed.

By turning **Continu** on, the views will be displayed continuously. After the last view is displayed, the sequence will be restarted. One can end the **RunViews** command by pressing the *End* key at any time.

Upon completion of the **RunViews** command, the **Drawing Window** will remain in the *perspective* mode for viewing. To return to *ortho*, you can exit the **Fly Thru** menu; select **3D Views** from the main menu of the "*ViewMast*" macro; and, then, select **Ortho**. Upon exiting from the **3D Views** menu, you will be returned to the main menu for the "*ViewMast*" macro.

A shortcut to return to the *ortho* mode of viewing is to select **DispVwes** from the **Fly Thru** menu. This not only returns to the *ortho* mode, but will also display the graphic representations of the temporary views which have been defined.

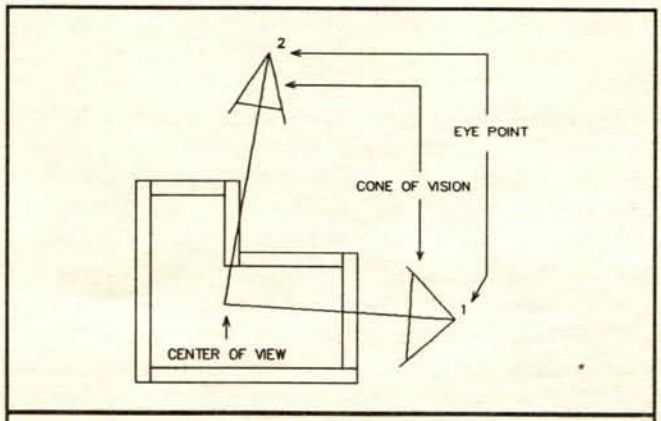


Figure 2
Description of Graphic Representation

CREATING 3D VIEWS FROM "FLY THRU" VIEWS

Temporary views can be added to the list of "*3D Views*" by selecting the **AddViews** command from the **Fly Thru** menu. All of the temporary views will be appended to the end of the list of the "*3D Views*". The order in which they are appended is the same as the order in which the views were defined.

Once the **AddViews** command has been selected, DataCAD will issue the prompt, "*Are you sure you want to continue?*". In addition, the number of views which will be created will be listed on the **Message Line**. Select "*Yes*" from the **Command Window** in order to continue. DataCAD will then issue the prompt, "*Enter a 1 to 6 character view name prefix:*". In response, enter up to six characters. Each of the "*3D Views*" created will have this prefix followed by a sequence number.

By establishing a naming convention, the prefix can be used to identify specific view sequences. For example, one could create the prefix by combining *view sequence*, *room number* and *view orientation*. Such a prefix might look like, "*A101SW*" in which "*A*", "*101*" and "*SW*" represent the view sequence, the room number and the view orientation, respectively.

Avoid using a prefix which was previously used with **AddViews**. The "*ViewMast*" macro does not investigate existing views to determine if the name has been used. As a result, using the same prefix more than once will result in views with duplicate names. This will result in confusion and accompanying problems down the road.

Once the **AddViews** command has been used, the temporary views are retained even after they have been added to the list of "*3D Views*". To clear them in order to define additional temporary views, you must select the **Reset** command. In addition, to clear the graphic representation of the views, you must press the "*Esc*" key.

Once the temporary views have been converted to "*3D Views*", one can then utilize the **AutoHide** function in the "*ViewMast*" macro to remove the hidden lines from the views.

THE "FLYALONG" FUNCTION

INTRODUCTION

The **FlyAlong** function provides another method for defining temporary views which can be converted to "*3D Views*". Many of the options or commands provided in **FlyAlong** are identical to those provided in the **Fly Thru** function.

The primary differences are the following:

Fly Thru: The user defines the eye and center points for each view.

FlyAlong: A *contour* entity is drawn which represents the *path* of the eye point. The center point will either lie along the contour or be established as a fixed point.

In **FlyAlong**, the temporary views are automatically created based upon the criteria with which the contour was drawn. Considerations for drawing the *contour* are identified in the following section.

DRAWING THE CONTOUR TO ESTABLISH THE PATH OF THE VIEWERS EYE

In order to develop temporary views with the **FlyAlong** function, one must first draw a *contour* entity. This entity can be drawn only when you are in the **Contour** menu. To access this menu, perform the following: Select **DCAD 3D**; then, select **3DEntity**; finally, select **Contour.**

A *contour* is defined by entering up to thirty-six "*control points*". After entering the necessary number of control points, DataCAD will draw the *contour* such that it passes through each control point.

There are several options on the **Contour** menu which control the smoothness and appearance of a given *contour*. These options were discussed in Vol. 3, Nos. 8 and 9 of *WindowIn on DataCAD*. The following briefly reviews the *primary* options which will affect the use of *contours* in **FlyAlong**.

Dvisions: This option controls the smoothness of the *contour*. The larger the number, the smoother the contour will appear. In effect, the number entered for **Dvisions** equates to the number of line segments used to draw the *contour* between each pair of control points. For the purposes of **FlyAlong**, the number of **Dvisions** identifies the number of temporary views which will be generated *between* each pair of control points defining the *contour*. Figure 3 illustrates this concept.

NOTE: *The documentation for the "ViewMast" macro provided with Version 4.0 of DataCAD states "DataCAD creates up to ten views for each segment of the contour." From testing, this limitation does not appear to be imposed. The number of temporary views seems to correspond to the number of* **Dvisions** *regardless of the fact that a value greater than ten is used.*

Z-height: Since the *contour* represents the path of the viewer's eye, the "*z-height*" at which the *contour* is entered into the drawing dictates the height for the eye point. With the

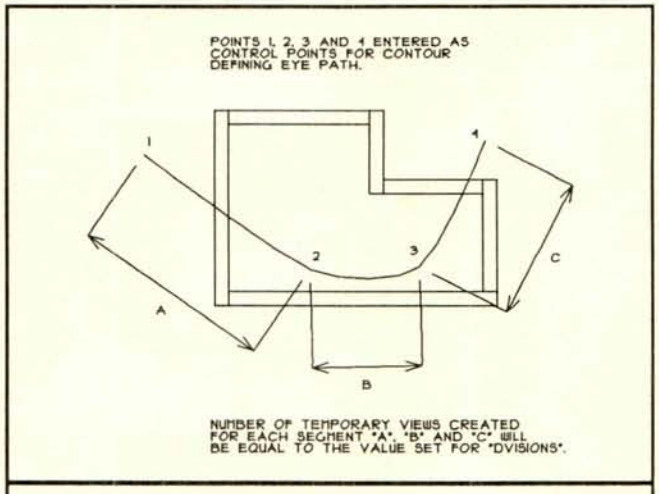


Figure 3
"Dvisions" with which Contour is Drawn Determines Number of Temporary Created

option **Fixed Z** turned on, the control points for the *contour* will be entered at a "*fixed Z-coordinate*". The value of the "*Z-coordinate*" can be entered by selecting one of the following options: **Z-Base, Z-User1, Z-User2** or **Z-Hght**.

With **Fixed Z** turned off, one can vary the "*Z-coordinate*" for each control point. The actual "*Z-coordinate*" used for each view would be interpolated from the "*Z-coordinates*" for the neighboring control points. As the views are generated, this would give the effect of the "*viewer*" looking up/down.

The type, i.e. **Natural, Cyclic** or **Tangent**, and the **Stiffness** of the contour will also affect the shape of the *contour*. For the purposes of the **FlyAlong** function, however, these options are secondary. For additional information on these, refer to Vol. 3, No. 8 of *WindowIn on DataCAD*.

OPTIONS FOR DEFINING VIEWS IN "*FLYALONG*"

Once the *contour* has been drawn to depict the path of the viewer's eye, the options in the **FlyAlong** menu can be established for defining the views. In contrast to **Fly Thru**, **FlyAlong** has only two options as described below:

ConeAng: Enables one to establish the "*angle of the viewing cone*". The angle entered restricts the amount of the model which can be viewed from the *eye point*. Specifically, with larger angles, more of the model will be visible. The value entered for **ConeAng** will be used for any subsequent temporary views defined. One can change the value for **ConeAng** as temporary views are defined.

FixdCtr: This is an on/off toggle. When turned on, DataCAD will issue a prompt requesting that you identify the center point of the views to be generated. **Center Z** will also be displayed in the **Command Window**, which, when selected, will allow you to set the "*Z-coordinate*" for the center point.

When **FixdCtr** is turned off, the center of the view will lie along the *contour*. Figures 4 and 5 illustrate the temporary views which will be generated with this option turned on and off.

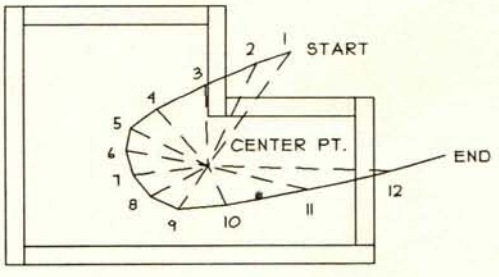
The **Delay** and **Continu** options pertain only to the display of the temporary views and function in the same manner as described for **Fly Along.**

Once **ConeAng** and **FixdCtr** have been set, the temporary views can be displayed by selecting **Begin**. DataCAD will issue the prompt, "*Select contour curve to define fly-along sequence.*" In response, use the mouse to identify the required *contour* curve.

DataCAD will then display the temporary views in accordance with the following:

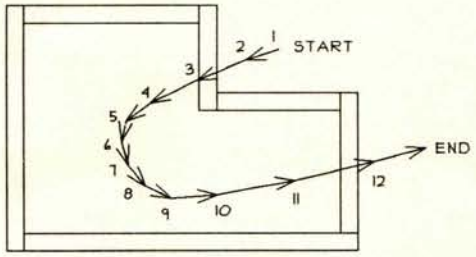
(1) The viewer's eye point will move along the *contour* following the elevation at which the control points were entered. Views will be displayed at each control point and at intermediate points based upon the number of **Dvisions** set when the *contour* was drawn.

(2) The "*focal point*" of each view will be defined by the setting for **FixdCtr**; and the "*breadth*" of the view will be defined by the angle established for **ConeAng**.



"FIXDCTR" TURNED ON AND NUMBER OF "DVISIONS" SET TO THREE RESULTS IN TWELVE VIEWS GENERATED ALONG CONTOUR LINE. ALL VIEWS "FOCUSED" ON THE CENTER POINT.

Figure 4
Views Generated with "FixdCtr" Turned On



"FIXDCTR" TURNED OFF AND NUMBER OF "DVISIONS" SET TO THREE RESULTS IN TWELVE VIEWS GENERATED ALONG CONTOUR LINE. THE CENTER POINT OF EACH VIEW IS "FOCUSED" ALONG THE CONTOUR LINE TOWARD THE NEXT "DVISION" POINT.

Figure 5
Views Generated with "FixdCtr" Turned Off

The sequence of views being displayed can be stopped at any time by pressing the "*End*" key on the cursor pad. Upon completion or ending the sequence of views, DataCAD will leave you in the *perspective* viewing mode. To return to *ortho* or some other viewing mode, you will have to exit **FlyAlong**. Then, you can select **3D Views** from the main menu for the "*ViewMast*" macro. Upon exit from **3D Views**, you will be returned to "*ViewMast*".

CONVERTING TEMPORARY VIEWS TO 3D VIEWS

The temporary views displayed in response to selecting **Begin** can be converted to "*3D Views*" by selecting the **AddViews** command from the **FlyAlong** menu. DataCAD will then issue the prompt, "*Select contour curve to define fly-along sequence.*" In response, use the mouse to identify the required *contour* curve.

The views will then be appended to the list of "*3D Views*". The order in which they will appear is the same as they would be displayed for the **Begin** command.

Once the **AddViews** command has been used, the commands in the 3D version of **GoToView** or their equivalents in the "*ViewMast*" macro can be used to rearrange their order or to delete unnecessary views. As stated earlier, the resulting "*3D Views*" can then be processed by the **AutoHide** feature of the "*ViewMast*" macro in order to obtain images with the hidden lines removed. If "*layer files*" are generated with these images, they can be used for developing a "*slide show*" using the **Slides** function in the "*ViewMast*" macro.

EXAMPLE SHOWING EYE PATH WITH VARYING Z-COORDINATES

As stated for both **Fly Thru** and **FlyAlong**, one can enter varying Z-coordinates for either the "*eye point*" or the "*center point*". When one generates views with a constant Z-coordinate, the views generated have a more "*robotic*" feel to them.

By varying the Z-coordinates, one can develop more realistic views in terms of how a "*person*" might look around as they walk through a building. Figure 6 illustrates a contour drawn for use with the **FlyAlong** function. Points 1, 5, 9 and 13 were entered as the control points for the contour. In addition, the Z-coordinate for points 1, 5 and 13 were entered at 4'; while the Z-coordinate for point 5 was entered at 8'.

The varying Z-coordinates will give the appearance of the viewer looking up as the path between points 1 and 5 is traversed. The viewer will then look down between points 5 and 9. Finally, the viewer will be looking straight ahead following point 9.

By varying Z-coordinates, **Dvisions** for the contour and **ConeAng**, one can generate more realistic "*walk throughs.*" After converting the temporary views to "*3D Views*", the commands in the 3D version of **GoToView** can be used to further edit the views.

WARNING

The settings and temporary views are "*remembered*" between times in which the "*ViewMast*" macro is used, *even though the macro may be used with different drawing files.* As a result, it is recommended that one initially use the **Reset** command prior to defining any temporary views. This will ensure that you are starting off with a clean slate.

Also, at certain times the prompt line will contain the prompt for the last operation performed and does not reflect the current menu. So, always review the prompt with respect to the menu/command that you selected.

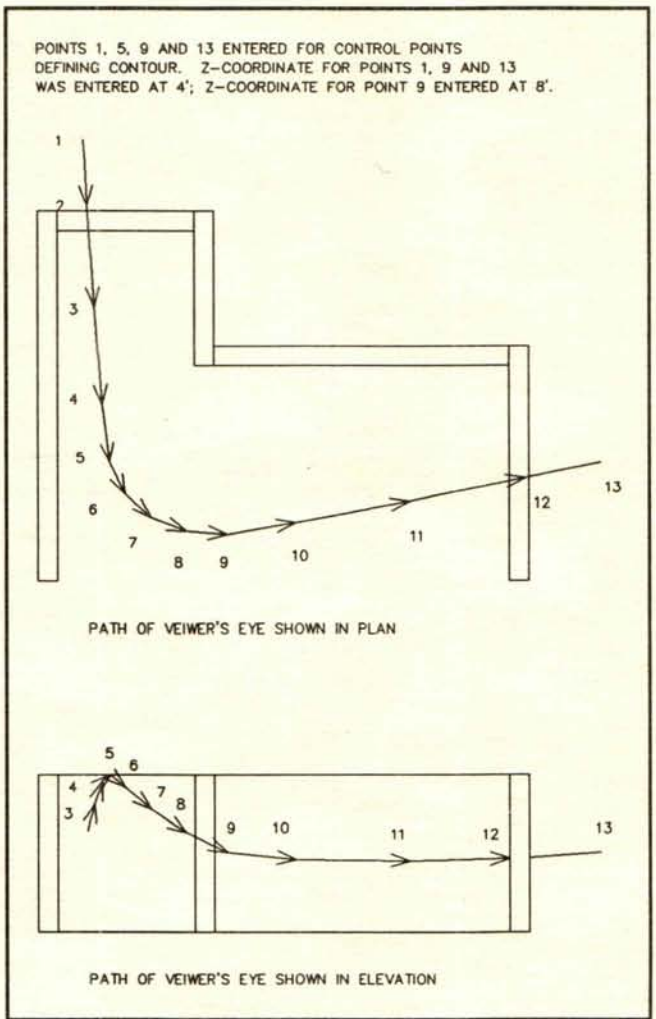


Figure 6
Entering Control Points of Contour
with Different Z Coordinates

WINDOWIN/EXTENTS

Phil Hart (CAD Manager, Moore/Weinrich Architects, Brunswick, ME)

A/E/C SYSTEMS SHOW

This year's A/E/C Systems Show in Washington, D.C. was the scene of the National DataCAD Users Meeting, hosted by CADKEY. The bulk of information presented and discussed is of interest to the DataCAD user. As a result, I have included a summary of it in this article.

THE PARTHENON PROJECT

Livingston Davies, cofounder and President of CADKEY, made a presentation focussing primarily on the direction of the long-term product development effort that has been underway at CADKEY since the acquisition of DataCAD. A product resulting from this effort will probably be released a year from now.

The effort, which was referred to as "*The Parthenon Project*," has focussed on the creation of completely new code written in C++. The new product (which was sometimes referred to as "*DataCAD 5.0*" and sometimes as "*Parthenon*") will initially be released to run under Microsoft Windows 3.x. For the sake of clarity about the forthcoming product, specific features listed in the side bar below were quoted directly from a CADKEY press release.

Livingston Davies went to some length to emphasize CADKEY's commitment to the continued development of the current (4.0) product. They will continue to enhance and market it in parallel with the new (Parthenon) product. Ultimately, the decision to discontinue the current product line will depend on the degree of acceptance of the new one.

CADKEY currently plans that when the Parthenon product is released, the DataCAD user (current on the Maintenance Agreement) will have the choice to continue with the DataCAD 4.0 product line or to switch to the new product at no additional cost.

Obviously, much of the energy at the users meeting was devoted to discussion of the new product: When would it be released? What would its features be? Who knows anything about Windows? What are the hardware implications? What does it all mean?

CADKEY is obviously doing something unusual in the software world in talking openly about a future product. They are opening themselves to a tremendous amount of criticism if they do not deliver the product by next year's A/E/C Systems Show. They recognize this and it is reasonable to expect that the DataCAD user should see something from them a year from now.

In terms of features, CADKEY people were somewhat elusive and unwilling to comment on features beyond speaking in generalities about the advantages of the Win-

Features Included in the Parthenon Project

- Full Windows support for multiple windows and multiple documents.
- An unlimited number of viewports.
- An unlimited number of drawing files can be simultaneously displayed.
- Multi-tasking operation for increased productivity.
- Protected mode 386/486 version for additional speed and access to virtual memory for larger drawing files.
- Full support for Dynamic Data Exchange so other programs can seamlessly and concurrently update data.
- Easy porting of AutoCAD third party products that support DDE to Parthenon.
- Full support for Dynamic Link Libraries to permit a much higher level of sophisticated third party products.
- The program is completely written in C++ to take advantage of modular, object-oriented programming and provides optimal integration of third party products as well as easier additions of new features to the program.
- Tighter integration of rendering and drafting that makes both functions easier to use together.
- Modularity in C++ ensures consistent operation across multiple hardware platforms.

dows environment. They did, however, acknowledge that they recognize the importance of the current product's ease of use and "*architectural*" character.

In terms of Windows and hardware, one of the advantages of writing the new product as a Windows application is that it frees CADKEY of having to write drivers for the program. From the user's point of view, if Windows can be configured, Parthenon will run. In conversations with Eric Smith (one of the programmers), it seemed to be clear that a reasonable minimum hardware platform would be a 386-25 with math co-processor and 4 megabytes of RAM.

In discussions with various people from CADKEY, it is clear that they are excited about the new product and its potential. They have a vision of this product as a "hub" for a complex suite of applications running under Windows and sharing a common set of data.

Much of its success will depend upon their ability to entice some of the big players in the AutoCAD third party world into the Parthenon environment. Since the new product is specifically designed to facilitate this (it has none of the technical, code-level road blocks that the current program does), and since CADKEY is placing a priority on accomplishing this goal, it is not unreasonable to expect that they will have some success in attracting third party applications.

One vision of architectural CAD is that the computer is a tool with which an architect "models" a project. Software applications enable the construction of a very complete model from which the architect can extract information in the form of graphics and data that ensure the accuracy and cost efficiency of the final built product. The process of generating, coordinating and communicating all of that information is immensely complicated. The approach that CADKEY is taking with the Parthenon Project will potentially move the DataCAD architect much closer to this vision of CAD.

OTHER NEWS FROM THE USERS MEETING

Softech

John Pegg, from the British branch of Softech gave a short presentation on Softech. Softech is the German distributor of *Spirit* (a slightly modified version of DataCAD). Their marketing approach is different than that in the U.S. A user typically purchases a turnkey system from Softech - hardware and bundled software - and receives technical support from them. *Spirit* includes optional DCAL macros that automate a number of DataCAD functions (principally in the modeler). Softech has its own development staff that writes and supports the macros and and *Spirit*, itself. CADKEY and Softech are currently exploring the possibility of marketing Softech products in the U.S.

Renderman

David Pendery, an architect from Cambridge MA, showed some slides of work that he produced using a DataCAD-to-Renderman interface that he is currently developing. This has strong potential for DataCAD.

"Reference Point"

I announced a new publication, tentatively titled *REFERENCE POINT*, to be distributed by CADKEY to DataCAD users. It will be mailed bimonthly as an enclosure with *3D WORLD* to all Users current on the Maintenance Agreement. It is intended to be a "*technical supplement*" that provides in-depth information to the DataCAD user. Its focus will be on technique, providing in-depth technical discussions of DataCAD functions within the context of suggesting strategies for accomplishing specific tasks. It will present information geared to users at all levels of experience with DataCAD.

Boundary Element Method

Peter Smith, cofounder and CEO of CADKEY, made a presentation on his latest research. He is developing software to perform materials analysis using "*Boundary Element Method*" techniques. This software will have no immediate impact on DataCAD, but may lead to products applicable to Civil Engineering.

AT THE BOOTH

CADKEY had a strong presence at this year's A/E/C Systems Show. Livingston Davies, Peter Smith, Eric Smith, Jim Melloy, Jeff Weymann, Lou Bodnar, Clay Rogers, Dale Arsenault and Mark White were all present at the DataCAD booth. In addition, Berry Taylor was on hand demonstrating DataCAD.

Third party developers were on hand demonstrating their products. See the *Extents* column for a report on third party people and products.

A final note from A/E/C: In the press release that announced the Parthenen Project, the following statement was made in the section regarding CADKEY's commitment to upgrading the 4.0 product: "*In addition, CADKEY,INC. also revealed that it has recently acquired the source code for Velocity, DataCAD's photorealistic rendering program, and plans enhancements to the program.*" As this was not discussed in Washington, this statement can only be taken at face value.

EXTENTS

This month, as *WindowIn on DataCAD* and this column come to a close, I want to summarize a few third-party macros that I have not reviewed. I will also discuss some products that I saw demonstrated at A/E/C Systems.

THE VINTAGE GROUP PRODUCTS

The Vintage Group, Inc.
8205 Lima Road
Fort Wayne, IN 46818
(219) 489-3543

AsciiMAX: $99.95
DuctPRO: $149.95
SteelPRO: $149.95
SubTEXT: $19.95
TxtLINE: $39.95

The DCAL macros from The Vintage Group perform a set of functions in DataCAD that range from the very simple to the moderately complex. I will describe each individually.

"*TxtLINE*" allows the user to create new, or modify existing, lines that contain text (similar to the T line as supplied with DataCAD). A line may be drawn on the fly with text interspersed at a user-defined interval. For the user who has the need for this function, "*TxtLINE*" works and is very reasonably priced.

"*SubTEXT*" is a "search and replace" macro that performs a very straight forward task. The user keys in a text string (up to 256 characters) to be searched and a replacement text string. Using the standard DataCAD editing options (entity, group, area...), the macro performs the task. Again, it works well and is inexpensive.

"*SteelPRO*" automates the drawing of steel shapes. It contains virtually all of the steel shapes for wide flanges, tees, channels, angles, tubes, and pipes currently listed in the AISC Manual of Steel Construction. Any shape may be drawn in plan, elevation, or section view.

In use, the macro is extremely simple. In the initial menu, the user selects the shape type. The next menu lists the sizes available. Having selected these, the user selects the view desired. Depending upon the shape and the view selected, different options are presented for the means of specifying the entry point/orientation of the shape to be drawn. This is an easy to use and accurate tool for accomplishing a specific task.

"*DuctPRO*" provides tools to an Architectural/Engineering firm for quickly drafting ductwork. The user specifies width and depth (or diameter), locates end points for the duct run and the macro draws it. Choices are presented for duct type (rectangular or round), sizes, Z elevation, elbows, and branches. The run may be drawn centered on the insertion points or offset to the left or right. Their Z elevation is relative to the specified Z height in DataCAD; the duct may be defined as having its bottom or top plane aligned to the Z height or the duct may be centered on it. Elbows, reductions, branches, vanes, and text are drawn as the insertion points are selected.

As a purely graphic tool, this is well implemented. A mechanical engineer might hope to see these functions as a portion of a more complex package which also performed all of the necessary calculations for sizing the duct work. But, given what it sets out to do, this is probably a real must for a mechanical engineer attempting to use DataCAD as a drafting tool.

I should note that I have used both "*SteelPRO*" and "*DuctPRO*" to draw exposed elements in 3D models. By using DataCAD's Z base and Z height controls, I was able to accurately draw steel shapes and duct runs with the macros. Using the **Explode/To Polygons** option in the modeler, the 2D entities were converted to polygons and used in 3D models.

"*AsciiMAX*" is a DCAL macro designed to facilitate the creation of schedules in DataCAD. The macro reads in *ASCII* files generated in a word processor and automatically places all of the text and graphic lines that make up the schedule.

The *ASCII* file must be very carefully composed to the requirements of the macro, which relies on the placement of TABS and empty lines to identify the relationship of elements in the final product. Once the macro has translated the *ASCII* file, it may also be used to edit the resulting schedule.

This strategy is similar to that used by desk top publishing packages. They allow the writer to imbed non-text *ASCII* codes in a word processor document that trigger formatting controls in the DTP program when the text is imported. The implementation of this strategy in "*AsciiMAX*" is fairly rudimentary, but works quite well. The biggest stumbling block to making it work efficiently is not a technical one but a management/personnel one. The person entering the original data in the word processor must adhere strictly to the conventions required by the macro. Overall, this is an interesting and potentially effective tool.

cadKEYNOTE

Integrated Systems
P.O. Box 10406
Raleigh, NC 27605
(919) 755-0117

Single-Use Version: $249.95
Multiple Site License Version: $299.95
MasterKeyList of Keynotes: $49.95
cadKEYNOTE PLUS (annual support service): $129.95

David Ward Jones was at the A/E/C Systems Show again this year demonstrating this macro. It is a fascinating one. I have had it installed on my system for nearly a year now, but have never really had a chance to use it to its fullest extent. As a result, I have not felt capable of giving it the sort of in-depth review that it deserves.

"*cadKEYNOTE*" consists of a DCAL macro and a large set of templates and symbols. The installation of the files is automated by an INSTALL.BAT file included with the product. The macro controls a process through which the user places symbols containing text and graphic information consistent with the Keynote annotation system. The symbol attributes are read by the macro as a means of linking every Keynote (placed in a .DC3 file) to a project specification. The macro recognizes the presence of spec sections required by the placed symbols and can flag sections of a master spec to be pulled to the project spec.

The intended user is one who uses both the Keynote annotation system and Con Doc master spec software. At Moore/Weinrich Architects, we rely upon an out-of-house spec writer and have never used a Keynote annotation system for our contract documents. Consequently, I have never been able to put "*cadKEYNOTE*" to work in a production environment.

Based on demonstrations of the product by David Ward Jones at this and last year's A/E/C Systems Show and on my own limited experimentation with it, I can make some observations about the product.

First, its features are very complete; the ability of the macro to search the active project spec to verify the presence of required sections (based on the "*cadKEYNOTE*" symbols placed in the drawing) is impressive. Its ability to coordinate between master and project specs brings a level of automation to an extremely tedious task.

Second, the portion of the macro that facilitates the creation, editing, and placement of the notational symbols is implemented in a way that facilitates the process. The menu structure is logical and the sequence of commands both efficient and easily understood.

Overall, the implementation of this product is excellent. The provision of support service (which includes product updates and a quarterly newsletter) is an important enhancement to such a complex product. The only fault that I can find with it is that it is so narrowly focussed that I, and many other DataCAD users, cannot take advantage of it. However, for an office which uses Keynote annotation on a regular basis and writes its specs in-house, "*cadKEYNOTE*" is an absolute must.

conTEXT

Modern Architecture
3130 Mayfield Rd. #211
Cleveland, OH 44118
(216) 321-0090

Price: $350.00

"*conTEXT*" is a DCAL macro that provides relatively sophisticated text handling capabilities to the DataCAD user. Installation is simple; the macro is copied to the \DCX directory. It is entered from the MACROS menu.

"*conTEXT*" sets out to provide a set of text formatting tools that are best described as being at the level of a basic word processor. That it accomplishes this in a graphical environment is a real achievement.

It incorporates switchable text profiles that are (optionally) plot scale sensitive, word wrap of text within user-defined column spacing, automatic underlining, case changing, note boxing, and arrow creation/placement. It has comprehensive File I/O features that allow text capture, import, and export; the user may print directly from DataCAD any text under the control of the macro. Its features allow for the creation of standard formats for schedules and legends, including the necessary line work. Text documents are maintained outside of the drawing file and are, therefore, accessible to multiple drawing files.

It took me a good bit of time to assimilate the meaning of terms used to define parameters within "*conTEXT*". However, once I understood the way that the macro was "*thinking*", its use was very straight forward. The macro's documentation is excellent, as is the provided tutorial.

Having familiarized myself with its use and having left it installed on my system, I found that I accessed it infrequently. Most of the text entry that I want to perform with DataCAD is accomplished in a simple fashion through the **Text** menu. I tend to handle text styling (font and sizing parameters) through "*TEXT-SET*" (reviewed December, 1990). As our specs are independent (of the drawings) documents, I have little call for the creation and placement of large chunks of text in typical working

drawings. For those occasions when large amounts of text are required, I have been in the habit of importing *ASCII* files generated by a word processor. Working on a 286-10, "*conTEXT*" is noticeably slow; when loaded to a 386-33, it ran at acceptable speed.

In the right setting, "*conTEXT*" could prove to be a very positive enhancement to DataCAD. I would imagine that an office working on projects of a scale that require spec language to be incorporated on working drawing sheets would find "*conTEXT*" to be a real boon. The independence of text from particular .DC3 files (text files are saved external to the drawing files) facilitates their transposition from one project to another. The ability to adjust the content and format of large groupings of text and the integration of the supplied tools into DataCAD make "*conTEXT*" a real step up from the word processor-sticky back routine that many offices use today.

THIRD PARTY PRODUCTS AT A/E/C

As mentioned elsewhere, David Ward Jones demonstrated "*cadKEYNOTE*" at the DataCAD booth. Fred Oesch, from Neo Graphix demonstrated his products (reviewed February, 1991), which I continue to be impressed with. Fred was also demonstrating "*Blocker*" for Bill D'Amico, who was unable to attend this year.

I met John Hitch who demonstrated his most recent macros for me. He has updated his double hung window macro, "*DHW40*" ($40.00) to include a much broader range of style options than the earlier version. He also demonstrated his "*PLOT40*" macro ($60.00), which I found to be very intelligent. It enables the user to lay out and control multiple scale plotting to a single sheet within a drawing file. The macro "tags" each layer, identifying the scale at which it is to be plotted. It generates grids aligned to the plot center that allow the user to accurately place the various pieces of the drawing relative to one another. Finally, and somewhat spectacularly, it allows the user to preview the finished sheet with all layers displayed at their correct size and placement relative to their own plot scales. These macros are available from:

Hitch & Associates
3309 Childers Street
Raleigh, NC 27612
(919) 782-4373

Tom Welsh, the developer of "*World View*", the forthcoming rendering package for DataCAD reported that he had received very positive feedback from users interested in his beta testing program. He anticipates that the product will be available in a release version in the Fall.

John Pegg, from Softech was in the booth demonstrating macros that they have developed for the German (and now British) market. The ones that I saw focussed on automating the construction of a 3D model based on a 2D plan drawing. While the interface for the macro (dual screen and somewhat confusing terminology) was unusual, the tricks that the macros performed were most interesting. Softech and CADKEY are investigating the possibility of bringing some of this work into the U.S., where I am sure that it will be favorably received.

IN CLOSING

In closing out this final article for *WindowIn on DataCAD*, I want to express my gratitude to a number of people. First, to the readers of *WindowIn* for their dedication to DataCAD and their constant willingness to share information about its use. Second, to the third party developers who have struggled to make DataCAD a more productive tool and struggled even harder to make a little bit of money in the process.

Finally, I want to thank Chris Davis for undertaking this newsletter in the first place. For a number of years, the information that he has distilled about DataCAD has been invaluable to me on a day to day basis. I also want to thank him for giving me the opportunity to write for the newsletter. It has been a remarkable experience.

NEW USER GROUP BEING FORMED -- NY

We have been notified about the formation of a new users group in upstate New York. For information, contact one of the following:

Tom Cisek, R. A.
(518) 842-3977
999 Design Group
25 McCleary Ave.
Amsterdam, NY 12010

Vito Mazzariello, R. A.
(518) 758-9046
999 Design Group
2 Skyview Lane
Valatie, NY 12184.