

Chapter Three

PROOF ANIMATION FRAMEWORK

The User Interface

Proof's menu system has six different Modes of operation: Run, Debug, Draw, Class, Path, and Presentation. Each Mode comprises a collection of closely related functions. Each Mode has its own unique menu bar and toolbar.

Overview of Proof Modes

Run Mode

Run Mode is the mode you use to run animations. You can start and stop the animation and control the speed and current time. If you give an animation to someone else for viewing without a scripted presentation, that person would use Run Mode as well.

Debug Mode

Debug Mode is similar to Run Mode, but adds tools for stepping through an animation and examining moving objects.

Draw Mode

Draw Mode is the mode you use to draw and edit layouts.

Path Mode

Path Mode is the mode you use to define paths along which objects can move. Paths are logical structures superimposed over lines and arcs.

Class Mode

Class Mode is the mode you use to define shapes that will be used as objects.

Presentation Mode

Presentation Mode is the mode you or someone else would use to control a scripted presentation. A presentation is one or more sequences of animation clips and static slides, possibly separated by special effects and/or controlled from one or more user-defined menus.

Input Files Used by Proof

Five types of files may be used as input to Proof Animation:

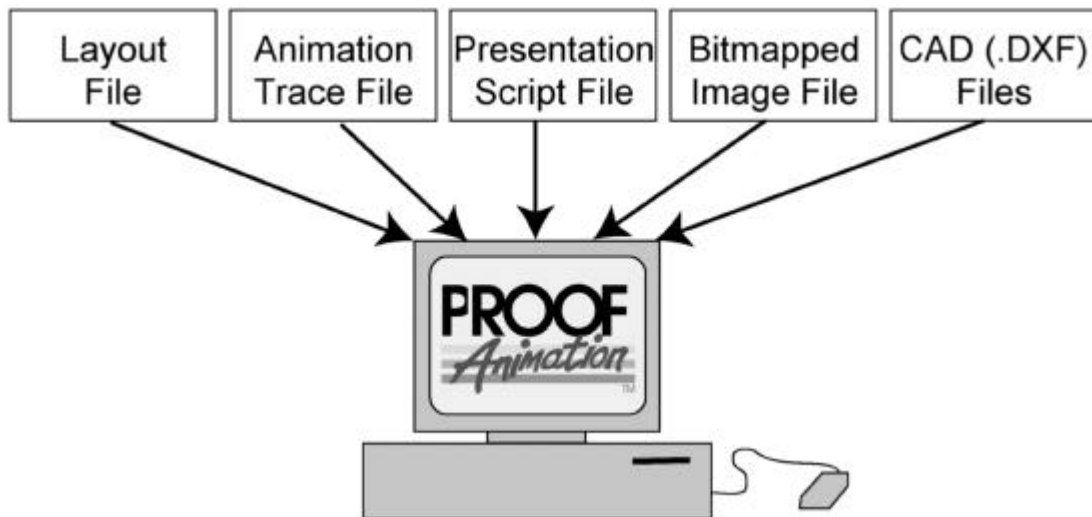


Figure 3-1. Five types of input files for Proof

Layout File (Required)

The Layout file contains all of the background text and graphics for an animation. It also contains definitions of Object Classes (shapes), Messages (variable text fields), Bar Graphs, Plots, and Paths (for guided motion). Layout files, normally constructed and saved from within Proof, may also be derived from data generated by other programs such as CAD programs (see Chapter 16).

In some cases, you might want to generate a Layout file from a program of your own. For this reason, the Layout file format is described completely in Appendix B. The Appendix B information is also available through Proof's online help system.

Animation Trace File (Required for Post-Processing Operation)

An ordered list of Animation Trace commands drives the animation. This list contains the sequence of timing information and other Proof commands that make the animation happen.

For post-processing operation (which is how most people use Proof), the Animation Trace commands are contained in an Animation Trace File. The Animation Trace File is generally written directly by a model (or program) that has been designed to generate syntactically correct Proof commands.

When Library (DLL) versions of Proof are used, animation trace commands are passed directly to Proof from a running program, and no trace file is required.

All animation trace commands are covered in a tutorial way in various chapters in this book. They are also described for reference in Appendix A. The Appendix A information is also available through Proof's online help system.

Presentation Script File (Optional)

A **Presentation Script File** consists of a sequence of commands that displays slides and animations. Proof has a special Mode for displaying the "slide show" described in a Presentation Script File.

Presentation Script Files are created by hand, using an editor.

The commands you can use in a Presentation Script File are covered in Chapter 17 and are specified in detail in Appendix C. The Appendix C information is also available through Proof's online help system.

Bitmapped Image Files (Optional)

Image files, when used for input, contain static slides usable only in Presentation Mode. The source of these files can be Proof (see below) or other software that supports the BMP or PCX file format. See Chapter 16.

CAD (.DXF) Files (Optional)

With a full commercial version of Proof you can directly import a standard .DXF-formatted CAD file and save the information in a Proof layout file.. The **Import .DXF** option is accessed by choosing **File, DXF Files**

Output Files

Proof Animation can produce six types of output files:

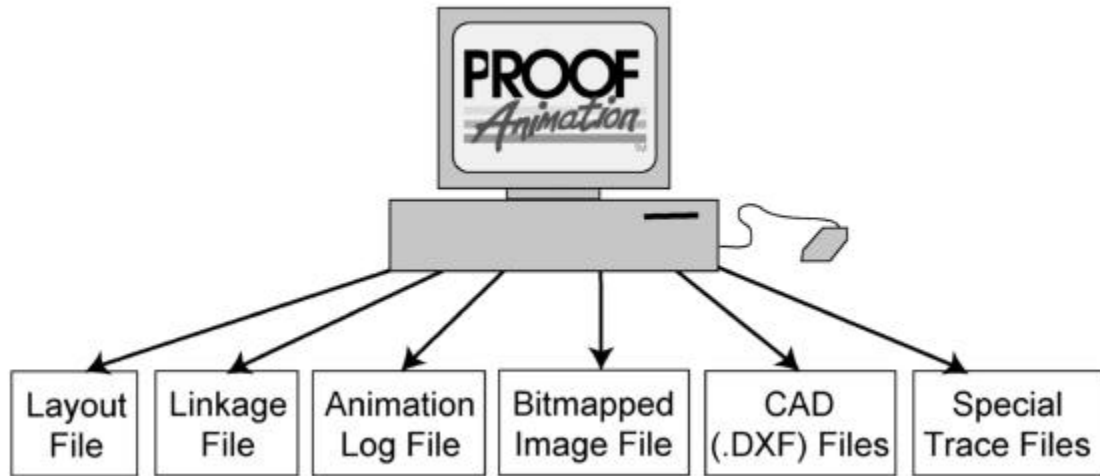


Figure 3-2. Six types of output files for Proof

Layout File

A **Layout File** is normally an *output* as well as an *input* file of Proof. At any time the current layout may be saved as a Layout File. Saving is typically done during and following a drawing session. (Proof does not include an auto-save feature.)

The Layout File contains geometry information that is generated in Proof or is input from a CAD drawing. The Layout File also contains various definitions and setup information.

Linkage File

From Draw Mode or Path Mode you can direct Proof to produce a Linkage file (see Chapter 16). A Linkage file contains descriptions of all Paths that have been defined in a Proof layout. Your model or program can extract important distance information from a Linkage file more easily than from a layout file. By having your model read this information from a file, you can be certain that time and distance information for Paths is consistent between your simulation and animation. Otherwise, each time you changed the geometry or speed of a Path within Proof, you'd have to manually update your simulation model, a potentially tedious and error-prone process. To generate a Linkage file, choose **File, Create Special Files, Write Linkage File**.

Log File

From time to time, Proof displays “alerts” — warnings, status alerts, or error messages — on the screen. More detailed information about the circumstances surrounding animation-related alerts is automatically recorded in a log file named *tracefilename.log*.

Bitmapped Image Files

It is possible at any point to save the current screen to a bitmapped image file for later use (see Chapter 16). You can do this by choosing **File, Grab Screen -> Slide**.

CAD (.DXF) Files

Proof layout files can be exported as .DXF files for use with CAD packages. You can export a layout by choosing **File, DXF Files, Export DXF**

Royalty-Free Demo Trace Files

All versions of Proof except Run-Time Proof can convert trace files into a format that allows them to be viewed with the royalty-free Proof Demo Viewer (or with Student Proof, used as a Demo Viewer). Proof does this by writing the encoding contents of an .ATF file into a .PTF file. The contents of a .PTF file cannot be altered. If the Demo Viewer detects modifications to a .PTF file, it will abort. To encode a Trace File, choose **File, Create Special Files, Encode .ATF -> .PTF**. .PTF files created with Student Proof are marked in such a way that Student Proof’s restrictions on layout file size, trace file size, and total viewing time are enforced, regardless of which version of Proof is used to view them.

Abridged Trace Files

Proof has the capacity to extract a portion of a Trace File. If you wish to show or distribute a small animation clip, you might want to create an Abridged Trace File in order to save space and processing time. If an abridged file starts at a time greater than zero, the initial portion of the abridged file is a sequence of commands that assigns all objects to their proper locations with their correct speeds. To generate an abridged Trace File, choose **File, Create Special Files, Write Abridged Trace**.

File Name Extensions

Proof uses the following file name extensions for input and output files:

.LAY	Layout
.ZZZ	Backup copy of Layout
.ATF	Animation Trace
.PSF	Presentation Script
.LKG	Linkage
.LOG	Proof Log
.PCX	Bitmapped Image (PaintBrush format)
.BMP	Bitmapped Image (Standard Windows format)
.PTF	Presentation (Demo or Run-time) Trace File
.XXX	Backup Copy of a Non-Abridged Trace File
.DXF	CAD Drawing File

On-line Help

Proof includes on-line help information for the commands a program can issue to direct Proof to perform required functions. For example, if you need to know the format of a Trace File command, you can find it quickly through on-line help. A Help option is always visible in the menu bar.

How Proof Measures Space: The Coordinate System

Proof uses a floating-point (real number) coordinate system for all geometry information. Animations can be created and viewed at any scale, and can be larger than a single screen. You can think of your screen as a “window” to a much larger “canvas.”

Proof measures space in floating-point “coordinate units.” One coordinate unit can equal whatever distance is most suitable for your animation.

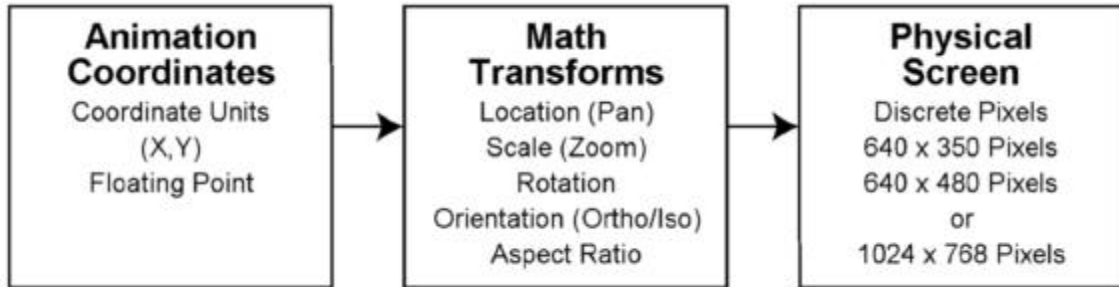


Figure 3-3. How Proof measures space

Most simulation models assume a dimensional unit such as inches or meters. It is usually convenient to match the coordinate unit in the animation to the dimensional unit in the model. Proof does not care whether your animation coordinate unit represents a centimeter, a foot, or a furlong. Whenever you start a new Layout, the screen you see will be 80 units wide unless you change the viewing scale. The point (0,0) is at the lower left hand corner of the screen in this “default Home View” (Figure 3-4).

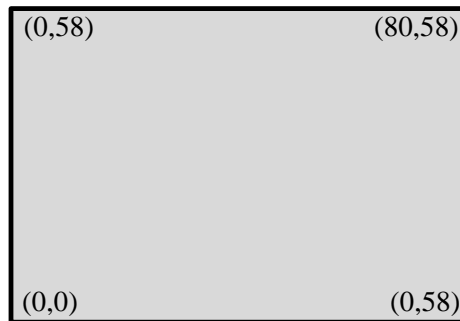


Figure 3-4. Screen coordinates in the default home view

You can change the default Home View for a particular Layout to be whatever you want.

It's easy to set the view to a different scale and location in the Proof coordinate system. You can do this from just about anywhere in Proof. You can save a particular view as the Home View so that the animation always starts with that view. You can save any view with a name for instant access later, and you can define as many of these “named views” as you want.

*For most animations, you will want to change the initial view that comes up when you start your animation (the Home View). This change must be made by choosing **View, Save Current View As**, and specifying the view name “Home” prior to saving the Layout File. (Saving the Layout File is discussed in Chapter 9.)*

How Proof Measures Time

Proof measures time in Floating-Point “animated time units.” One time unit can equal whatever duration is most suitable to your animation.

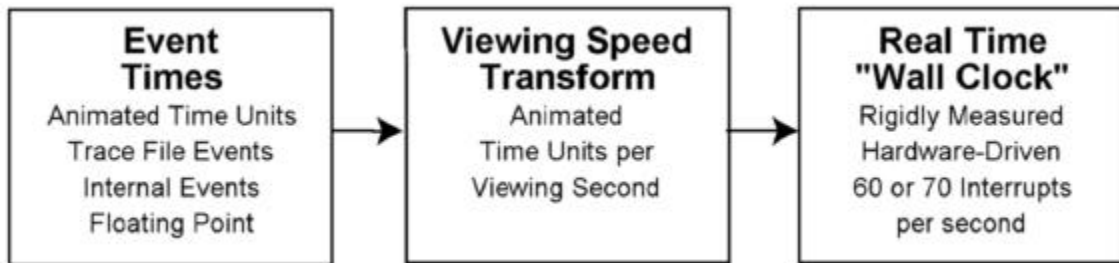


Figure 3-5. How Proof measures time

An animation runs at a particular “animation speed.” This speed is displayed on the screen in *animated time units per second of viewing time*. For instance, if your simulation and animation use seconds as a unit of time, and the Proof speed is 1.00, then your animation is running at one animated second per viewing second. You can change the speed of a running animation at any time.

Proof does not care whether one time unit represents a microsecond, a minute, a day, etc. The animated time unit you choose should match the time unit used in the model or program that is driving Proof.

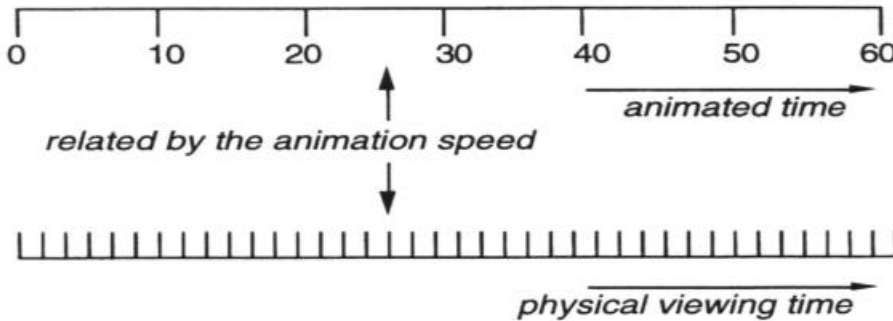


Figure 3-6. The relationship between animated time and wall clock time

The default animation speed in a new animation is set at 6 animated time units per viewing second. You can change the default speed for any animation.

The animation speed is stored as part of the Home View. In order to permanently change the starting animation speed for a particular layout, you must first change the speed to the desired value using the Run Mode menu bar, then update the Home View, and then save the layout.

Navigating Proof's User Interface

Invoking Menu Options

Proof follows standard conventions to invoke menu functions. Menu items are selected by clicking on them with the left mouse button, or for items containing an underscored letter, by typing the letter with the **ALT** key pressed. For example, select the “File” menu item by typing **ALT-F**. Items that contain underscored letters in pull-down menus can be selected by typing that letter. For example, in the File menu, **E**xit can be selected by typing the letter “x.”

Each menu option is explained in greater detail elsewhere in this book.

Keyboard Shortcuts

The following keyboard shortcuts are available for frequently invoked functions:

- <Ctrl-C> - Invokes the **Select Class** dialog
- <Ctrl-L> - Invokes the **Open Layout** dialog
- <Ctrl-O> - Invokes the **Open Layout & Trace File** dialog
- <Ctrl-P> - Invokes the **Select Path** dialog
- <Ctrl-S> - Invokes the **Save Layout** Dialog
- <Ctrl-T> - Invokes the **Time Jump** dialog (in Run/Debug/Presentation Modes)
- <Ctrl-V> - Invokes the **Select/Delete Named View** dialog
- <Ctrl-W> - Invokes the **Active Window Properties** dialog
- <Home> - Switches to the Home View
- <Numeric Keypad +> - Run Mode: Starts/resumes an animation
- <Numeric Keypad +> - Debug Mode: Executes the next trace command
- <Numeric Keypad +> - Presentation Mode: Executes the next presentation step
- <Numeric Keypad +> - Class Mode: Invokes the **Add Class** dialog
- <Numeric Keypad +> - Path Mode: Invokes the **Add Path** dialog
- <Numeric Keypad -> - Presentation Mode: Goes back to the previous presentation step

Mouse Wheel Shortcuts for Panning & Zooming

If your mouse has a wheel, Proof will recognize the following shortcuts for panning and zooming:

Keyboard State	Wheel Rotation	Action Taken
Normal	Toward you	Pan down (layout moves up)
Normal	Away from you	Pan up (layout moves down)
Shift pressed	Toward you	Pan right (layout moves left)
Shift pressed	Away from you	Pan left (layout moves right)
Ctrl pressed	Toward you	Zoom in
Ctrl pressed	Away from you	Zoom out

