POSER® Custom Model Maker's Guide

by
Dave Hill

Preparing a Base Model

An exercise on customizing figure models used by Poser 3.

Compiled March 1999 by Dave Hill
www.aarrgghh.com/poser
dave@aarrgghh.com

Introduction

You can't imagine my excitement when Poser 3 finally hit the stores. The latest version boasted models of higher resolution and detail than ever before, better animation controls, and most exciting of all, the ability to create my own Poser models. No longer would I be restricted only to the figures supplied by MetaCreations. I envisioned animating an entire pantheon of custom models, human and animal.

Unfortunately, this last promise proved to be just that, a boast.

First of all, nothing of the sort was mentioned about custom models in the relatively short 200-page manual supplied with the program. A readme file on the program disc directed me to the MetaCreations' web site, where I found the relatively shorter 75-page Adobe Acrobat PDF format *Poser 3 Advanced Techniques Guide**. While the document did indeed provide information on creating custom models, it was far from complete; there is simply no way anyone could take a model from start to finish and expect it to work in Poser. Without a thorough explanation of the formats of the geometry and resource files used by the program, the document's little better than a hint sheet.

MetaCreations' tech support provided less help than the book: when I asked if any sort of technical white paper existed on Poser models and their resource files, I was instead directed to seek the advice of online user forums. In other words, I was on my own.

Undaunted, I took up the challenge, and after three weeks of tedious and repetitive trial-and-error experimentation and meticulous recordkeeping and chart-making, I made a breakthrough, and for the first time realized that it is in fact possible for mere mortals to create fully operable custom models for Poser. Almost another two weeks passed before I successfully completed my first model. But my work was not yet done.

Now that my head was swollen with all this undocumented technical data, the next task was to set it to type in an easily understandable format before I started to forget it all. All of a sudden I was faced with producing a tutorial of undetermined length in as short a time span as possible; for all intents and purposes I was back in college cranking out a term paper. Ugh!

Nonetheless, another month and countless screen captures later, the *Poser 3 Custom Model Maker's Guide* was finally finished. Given the speed with which I crammed this together, I should really call this a beta version. So the ball's in your court now—it's time to put this guide to the test. I hope I've filled—at least partially—a massively gaping hole in the documentation for this otherwise wonderful program. I hope I've opened the door for more Poser enthusiasts to take their projects to the next level. Most importantly, I hope I've significantly cut down some of the time it takes to complete this inherently tedious and time-consuming process.

Whom This Guide is For. This guide is not intended for absolute beginners and assumes you understand general 3d modelling principles and techniques, and have some experience creating models and manipulating 3d wireframes. It also assumes you know your way around both Poser and Ray Dream Studio. Though I have attempted to be as detailed as possible, this book is less a step-by-step elaboration on technique than an overview of the process of creating models designed to work with Poser 3. Therefore, this book is not intended to replace the Poser or Ray Dream manuals or the *Advanced Techniques Guide*, so you should keep them handy.

This document does not cover creating models completely from scratch but describes how to customize the high-resolution human figures supplied with Poser. Until my own guide is finished, a solid online tutorial on starting from square one, for figures human and non-human, can be found at

**www.metacreations.com/products/poser3/advanced_techniques.pdf*

www.spiritone.com/~bushi/tutorial1.htm

Other documents and developments on this subject can be found online at the Poser Forum Online (www.iguanasoft.com/poser). Commercially available guidance in this area can be found in Shamms Mortier's *The Poser 3 Handbook* (Charles River Media, 1999).

This guide covers modelling only on the Macintosh platform; Poser files on the Windows platform have a slightly different configuration. Hopefully, the techniques covered in this guide are broad enough to be nonetheless applicable to those users.

Though the bulk of the modelling tasks described in this book are accomplished with MetaCreations' Ray Dream Studio, I have tried to emphasize the goals of each step, in order to help the users of other modelling programs adapt the tools of their preferred programs to each task.

In fact, the models in this guide were constructed with a variety of 3d programs, including NewTek's Inspire 3d (the younger sibling of the venerated LightWave). Where appropriate, I have described tasks utilizing the tools of these programs.

Terminology. As this guide was being written I fell into using certain 3d terms synonymously and want to make this clear to the reader:

Point, vertex: Any single location in space described by one set of 3d coordinates.

Edge, line, link, segment: Any unclosed succession of two or more linked vertices

Face, facet, polygon: Any closed group of three or more linked vertices.

Lattice, mesh, polymesh, wireframe: One contiguous group of polygons; a 3d object.

1. Collecting Our Raw Materials

In this exercise we'll be customizing Poser's default male nude model in order to create a basic model from which we can create a superhero. Most superheroes wear gloves and boots, so we'll need to alter the model's hands and feet. Also, while most superheroes strut about in impossibly skin-tight leotards, a model with stark nipples and a bellybutton might be a bit much, so we'll be removing those.

Let's identify the Poser files you'll be working with. Every Poser model comes with at least two files. The first is the geometry file, which contains the actual 3d data about the shape of the figure. Poser's geometry files are in the Alias Wavefront OBJ format (an ordinary text file that can be opened by any word processor) and are be found in the `Geometries` folder. This folder sits in Poser's `Runtime` folder, where the program stores all its various resource files. Specifically, the Nude Male 3d file sits in the `newMaleNudeHi` folder and is named `nudeMaleWithEyes.obj`. See Figure 1.

In the same folder you'll notice the file `newHipNoGen.obj`. This is the alternate geometry file for the hip, which features no genitalia. When deselecting the Genitalia option on Poser's Figure menu, the program substitutes this file for the hip object in `nudeMaleWithEyes.obj`, which by default features genitalia. Since most superheroes, despite their egos, are particularly modest in this respect, we'll be needing the alternate hip file as well.

Poser uses a resource file to manage the figure's geometry, which defines the separate working parts, their kinematic relationships, and the textures used to paint the parts. This is also an ordinary text file identified by its `.cr2` extension and found in the `Runtime:libraries:characters` folder. It is actually a program, which contains function calls that instruct Poser on manipulating the geometry. Specifically, we'll be using the `Nude Man.cr2` file in the `Runtime:libraries:characters:People` folder.

Poser's PC version actually splits the resource file into two parts, the .cr2 file and the `.rsr` file, both found in the `Runtime:libraries:characters` directory. The `.rsr` file contains the small library preview graphic icon, which on the Macintosh version is integrated with the `.cr2` file.

To reiterate, the files we'll be working with, (using their full paths names) are:

Geometry files:

```
Poser:Runtime:Geometries:newMaleNudeHi:nudeMaleWithEyes.obj
Poser:Runtime:Geometries:newMaleNudeHi:newHipNoGen.obj
```

Resource files:

```
Poser:Runtime:libraries:characters:People:Nude Male.cr2
Poser:Runtime:libraries:characters:People:Nude Male.rsr (PC Poser only)
```

2. Setting Up Our Workspace and Files

Let's create a new folder in the Geometries folder and call it `myHero`. This is where we'll be storing our new geometry file. Copy the geometry file `nudeMaleWithEyes.obj` and move the copy into the `myHero` folder. Rename your new geometry file `myHero.obj`. See Figure 2.

Since Poser automatically stores new figure resource files in the folder `Runtime:libraries:characters:New Figures`, we'll simply duplicate the files `Nude Man.cr2` and `Nude Man.rsr`, rename them `My Hero.cr2` and `My Hero.rsr`, and move them into the `New Figures` folder. Again, the `.rsr` file is associated only with the PC version of Poser; Mac users need not be concerned with this file, since it is already part of the `.cr2` file.

3. Cleaning Up Our Resource File

Now we need to edit the resource file in order to change its references to the original nude male geometry to our `myHero.obj` geometry file. We'll also remove all references and function calls to the genitalia that "My Hero" will no longer need. Note that for pathnames Poser uses the Macintosh convention of using semicolons to indicate directory levels.

Figure 1: Their Files

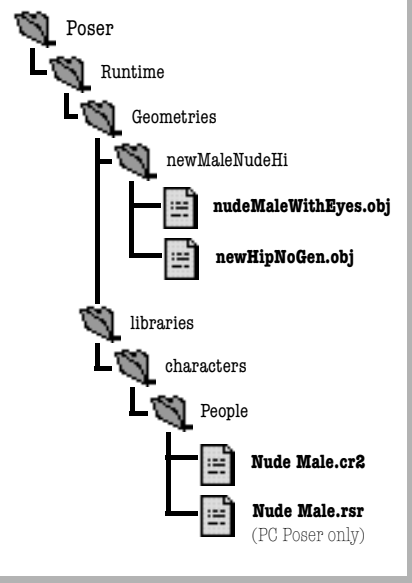


Figure 2: Our Files

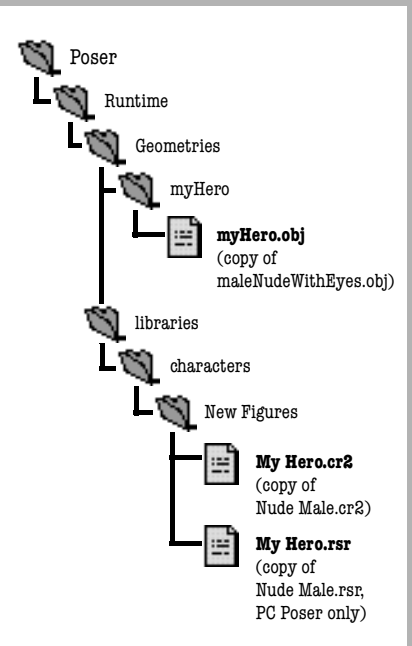
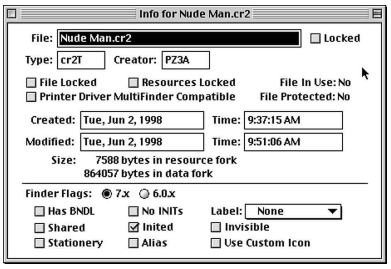


Figure 3: Macintosh File Types and Creators



Can't find our resource file `My Hero.cr2` listed among the other figures in the Poser library? Presuming we've saved it in the right place, our text editor may not have saved the file with its original type and creator codes. Without the proper type/creator codes, Mac Poser will not recognize the file as a legitimate library resource.

Open ResEdit or any other such program useful for poking about in the resource fork of Mac files. Select "Get File/Folder Info ..." from the File menu and open `My Hero.cr2`.

Enter `cr2T` for the file's type and `PZ3A` for the creator code. Save and exit. The file will now show up in Poser's figure library.

Open the file `My Hero.cr2` in a text editor. Replace all occurrences (there are two) of

```
figureResFile :Runtime:Geometries:newMaleNudeHi:maleNudeWithEyes.obj
```

with

```
figureResFile :Runtime:Geometries:myHero:myHero.obj
```

To eliminate the genitalia from the resource file:

Delete `actor genital` object declaration (5 lines)

```
actor genital:2
    {
        storageOffset 0 0 0
        geomHandlerGeom 13 genital
    }
```

Delete `alternateGeom` subroutine from parameter settings function `actor hip:2` (5 lines)

```
alternateGeom    hip_1
    {
        name Genitals Off
        objFile 20 :Runtime:Geometries:newMaleNudeHi:newHipNoGen.obj
    }
```

Delete 3 relative movement subroutines from parameter settings function `actor hip:2` (93 lines)

```
twistY genital_twisty
    {
        statement
        statement ...
    }
jointZ genital_jointz
    {
        statement
        statement ...
    }
jointX genital_jointx
    {
        statement
        statement ...
    }
```

As well as this relative scaling subroutine from parameter settings function `actor hip:2` (28 lines)

```
smoothScaleZ genital_smooZ
    {
        statement
        statement ...
    }
```

Delete all of the parameter settings function `actor genital:2` (402 lines)

```
actor genital:2
    {
        name    genital
        statement
        statement ...
        customMaterial    0
    }
```

Delete grouping function `addChild` (2 lines)

```
addChild    genital:2
    hip:2
```

Delete stitching function `weld` (2 lines)

```
weld    genital:2
    hip:2
```

Be very careful when editing this file. Mistakes may cause unpredictable results in both our model's behavior in Poser.

We're done! Save `My Hero.cr2`

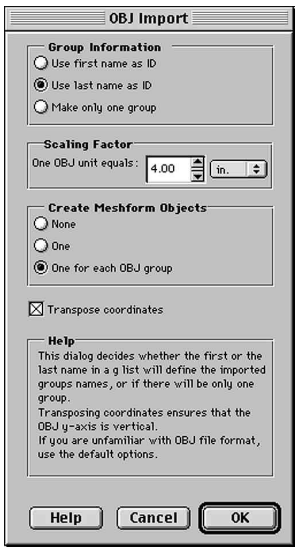


Figure 4: Ray Dream Studio OBJ Import Dialog

4. Editing MyHero's Geometry

Let's open the geometry file `myHero.obj` in our favorite modelling program. If the modeller does not open OBJ files directly, but uses an import utility or plug-in, we'll probably be faced with at least two options, for scaling and grouping. Do not change the importer's default setting for scaling. If `myHero.obj` imports very small or very big, adjust the modeller's cameras and/or view magnification, but do not change the model's size or position, especially the position.

If we must change its scaling, use a simple and memorable multiplier like 10 or 20, because we'll need to restore the model's original dimensions when we're done and ready to save the edited OBJ file. It's also a good idea to jot down the model's origin point before changing the scaling, because this may change during the scaling and/or editing processes. It's crucial than when we're done the model has the same size and position it came in with.

Make sure the option for grouping is selected so that the model imports as a series of distinct named parts instead of a single object. Some modellers import OBJ files as a single object with distinct named surfaces; this may be perfectly fine as long as it can export the surfaces as a series of grouped polygons named by bodypart, following Poser's naming conventions. You may have to rename the surfaces yourself in order to get them to conform to Poser's conventions.

Delete myHero's hip and genitals. Open `newHipNoGen.obj`. If myHero was scaled in any way, scale the hip object accordingly and copy it to memory. Paste the hip object into myHero; it should pop perfectly into place. If necessary, change both the hip's object name and master name to `hip`. Close `newHipNoGen.obj`. Export `myHero.obj`. in Wavefront OBJ format. (When exporting, make sure to always open the **Options** dialog and select **Full path for grouping**.) Remember to undo any scaling of the figure before export.

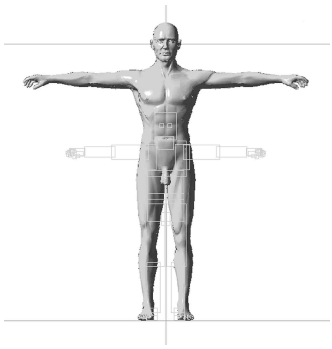
I should note that I prefer to save the file first in my modeller's native file format, as `myHero.rds`, and export copies in the OBJ format used by Poser. This way I get to preserve any modelling features specific to the program. If I need to rework an extrusion or lathe operation, I'll still have access to the original outlines or curves used, which would be otherwise lost in the conversion to a foreign file format. Thus we edit and save the native file, then export the OBJ copy for Poser. ***Thus from this point on when I refer to opening*** `myHero.obj`***, I mean opening the native file*** `myHero.rds`***. When I refer to saving*** `myHero.obj`***, I mean saving the native file*** `myHero.rds` ***and exporting the OBJ copy*** `myHero.obj`**.** (Just trying to save myself some typing ...)

We can check our progress immediately by opening Poser, and selecting **My Hero** from the **New Figures** library. What should appear is the myHero model, which for now is only an emasculated version of the original **Nude Man** figure, and which may or may not have retained its original colors and textures.

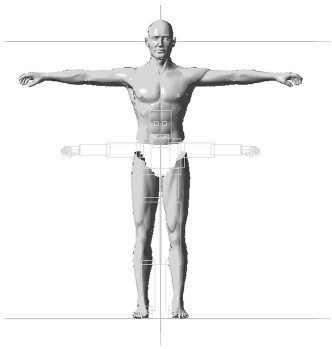
5. Where's My Hero?

At this first stage in the game, we've set up our model and we're ready for further editing. However, we may have run into one or two problems on the way. I'll quickly

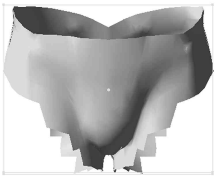
Figures 5-8: The Unkindest Cut



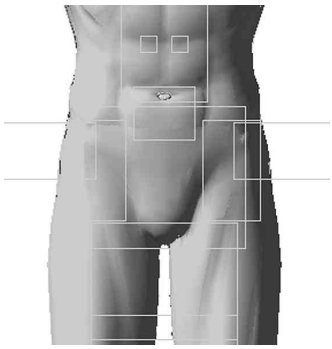
Open myHero.obj ...



...delete hip and genital objects ...

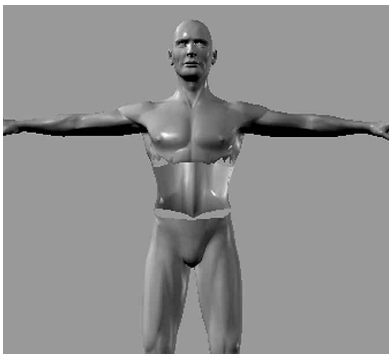


...open and copy newHipNoGen.obj ...



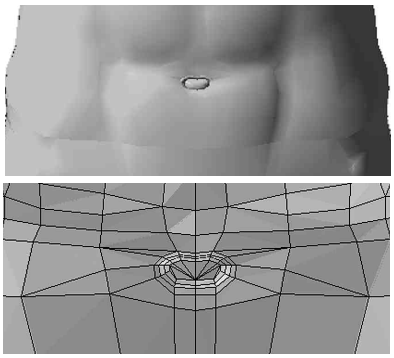
...paste into place and save file.

Figure 9: Meet Abby Normal

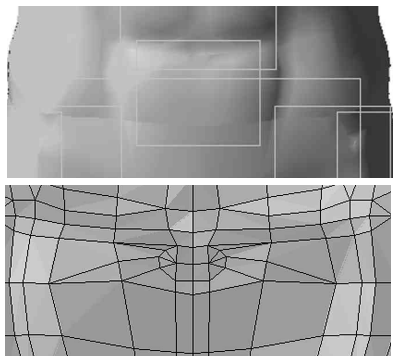


If any of our Hero's body parts disappear, or appear inside-out,as does this abdomen, we've got a case of reversed surface normals ... Poser 3.0.1 intends to better address this problem.

Figures 9-10: Tummy Tuck



The patient prior to surgery ...



...and after, just a few facets lighter.

address two here, since we'll run into them again each step of the way if they're not resolved.

First, if Mac users are having problems with My Hero not appearing in Poser's **New Figures** library and the resource file `My Hero.cr2` is in the **New Figures** folder, our resource file may not have been saved with the proper file type and creator codes by our text editor. These codes are used by the Mac OS to identify the program any file may belong to. This is how the system is able to determine which program to launch when we double-click a document file. Some programs, like Poser, use this information to display only its own file types in its dialogs, presumably to limit the visual clutter of long file lists to legitimate Poser documents.

This problem's easy to solve if you've got a copy of **ResEdit**, Apple's free resource editing utility. If you don't, the utility is readily available from Apple's download site (`ftp://ftp.support.apple.com/apple_sw_updates/US/Macintosh/Utilities`), America Online, and many online softwarte libraries.

Open **ResEdit**. The first thing ResEdit will do is ask for a file to open; locate `My Hero.cr2` and open it. Select **Get Info for My Hero.cr2** from the **File** menu. In the **Info Dialog** (see **Figure 3**) enter `cr2T` in the **Type:** field and `PZ3A` in the **Creator:** field. That's it. Save and quit. Be aware that we may have to do this several times over the course of the project, especially if our text editor insists upon saving files with its own codes.

5. Who's Abby Normal?

Our second possible problem concerns the geometry file. We've successfully loaded My Hero into the **Figure** library. A simple double-click loads the figure into the main window. Uh-oh. One or more of his parts appears missing or inside out. We may have a case of reversed surface normals. See **Figure 9**.

Surface normals are associated with each polygon that makes up a 3d object and describe the direction that each polygon faces. In some 3d modellers, normals are represented by a line perpendicular to the polygon face; it may point away from the object, indicating an outer surface, or it may point into the object, indicating an inner surface. Normals are a lighting aid, used by 3d rendering programs to determine how to portray the surface's orientation by creating the proper highlights and/or shadows for it.

Ray Dream Studio does not display normals, so we can never tell if they're correct until the model's rendered. **Inspire 3d Modeller** displays normals, and also allows us to render polygons as ***double-sided surfaces***, which eliminate rendering problems caused by reversed normals. Double-sided surfaces are polygons with two outer surfaces and no inner surface—unlike ***single-side polygons***, which have an outside and a normally invisible inside surface. Because Ray Dream's models usually render properly, it is safe to assume this program also renders all polygons as double-sided.

Poser 3.0 obviously renders polygons as single-sided; to correct our model's normals, open the geometry file `myHero.obj` in our modelling program. In **Ray Dream**, select and **Jump Into** (Ray Dream's term for launching a modelling environment) the culprit bodypart (in the case of **Figure 9**, the abdomen) with the **Mesh Form Modeller**, Ray Dream's vertex-level editor. From the **Edit** menu select **Select All** for the entire mesh. From the **Polymesh** menu select **Reverse Polygon Normals**. In **Inspire 3d Modeller**, we'd use the **Align** command in the **Tool** tab. But that's all folks. Save `myHero.obj` and quit.

(Oddly, Ray Dream does not record this action as a change in the file; hence the **Save** command in the **File** menu remains dimmed and inaccessible. Simply select **Save As ...** from the **File** menu and overwrite the file by keeping the name.)

Note: **Poser's 3.0.1 update**, released 1/12/99, is supposed address the problem of inconsistent normals by introducing additional user options for flipping normals on import. And the update will now read any normal data contained in OBJ files. So this section may be now entirely moot.

6. Flatten That Tummy!

Eliminating the model's bellybutton, part of the abdomen, is a relatively easy task of deleting polygons and creating new patches over the ones deleted. See **Figures 9-10**. Any 3d program that can import the geometry should be up to the task. While studying the pictures provided in this tutorial, you should feel free to improvise, especially if you

feel you can do a better job. We can make the job as easy, detailed or difficult as we please.

No doubt for most of you absolute beginners out there, the first and most difficult skill you'll need to develop will be simply selecting, from the hundreds that comprise any of these models, the exact vertices, edges, and polygons you want to operate on at any given moment. This often involves patience and agile viewpoint-switching.

For all vertex-level editing tasks presented within these pages, the general strategy I recommend restricts itself to removing only one or two polygons at a time. We must be careful not to create a situation in which we're forced to add vertex points in empty 3d space in hopes of linking it to one or more existing ones. It is sometimes difficult to judge our position in empty space and attempting to do so requires repeated changes in viewpoint, which can easily make an already inherently tedious process ever more tedious. In editing these models I've never added a single free point—instead I've linked existing ones and subdivided the resulting segments to provide additional vertices, or deleted unwanted ones and linked the remaining vertices to reduce detail.

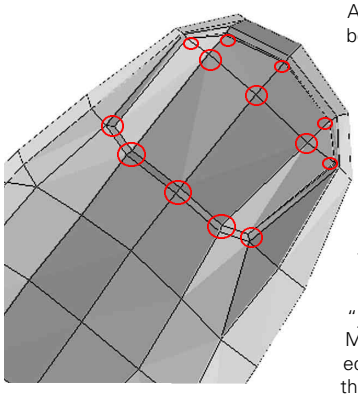


Figure 15: Part lThumb3 and vertices to be removed

After all, we're customizing a model that's been supplied to us and 99% of our work is therefore already done—we must use the design of the model's own wireframe to guide our reconstruction.

Ray Dream Studio's **Mesh Form Modeller** sports a polygon decimator which automatically deletes polygons by number or percentage, but you may find, as I have, that its results are too unpredictable to be useful on a model of this level of detail.

To take care of My Hero's belly, simply "jump into" the abdomen in the Mesh Form Modeller and delete the all but the outermost edge of the facets defining the bellybutton. Link the vertices across the edges of the resulting hole and use the **Fill Polygon** command to create the polygons that will seal it. (In Inspire 3d the analogous command is **Make Polygon** in the **Tool** tab.) And that's all, folks.

Note: If the links don't completely fill, we have one or more improper links. A surefire way to determine whether our edges are proper is to select the polygon they're attached to. A proper edge will become selected as the polygon is selected; an improper edge will not become selected. For example, if we click on a three-sided polygon and only two sides become active, the inactive edge reveals that one of the endpoints on that edge is not linked to the third.

7. Nip and Tuck

Taking care of the nipples, which are found on the model's collar parts, can be even simpler. In this case I've merely pushed in the facets describing the nipple in, to surface level. See **Figures 11-14**.

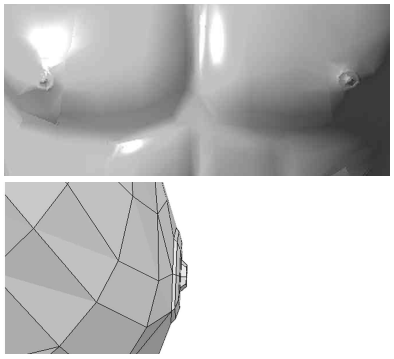
The two parts to be edited are named `rCollar` and `lCollar`.

8. Kid Gloves

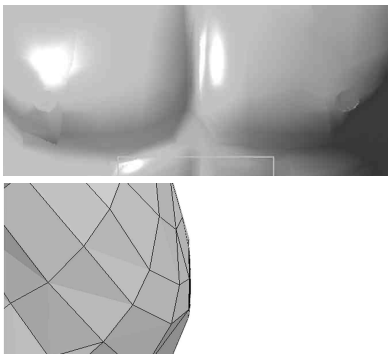
Removing the figure's nails, all ten of them, will prove to be the second most time-consuming task we've attempted yet in this tutorial. Everything I've said regarding our approach to remodelling the bellybutton applies to this task. The idea again is to eliminate the facets comprising the nails, and to use the wireframe lattice as our guide in remodelling, in order to remain faithful to the model's general shape. Study **Figures 15 through 17** to get a feel for your approach. While the speed of your work will increase as you hone your technique from finger to finger, there's no point in rushing the job, because you'll hate having to do it over again. I've repeated this chore quite a number of times already in the process of teaching myself, so believe me, you don't want to have to do this any more times than you have to.

The ten parts to be edited are named `rThumb3, rIndex3, rMid3, rRing3, rPinky3` and `lThumb3, lIndex3, lMid3, lRing3, lPinky3`.

Figures 11-14: Breast Job

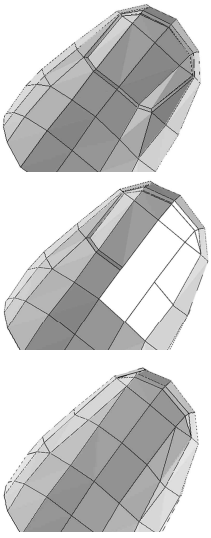


Now you see 'em ...



...now you don't.

Figures 16-18: A Manicure

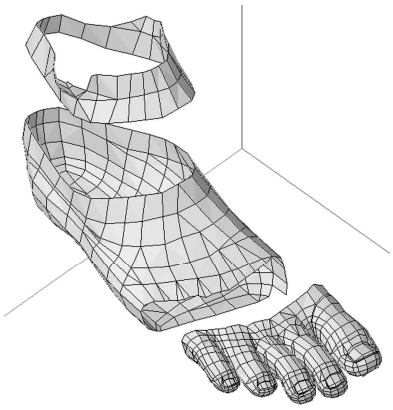


Original Poser 3.0 thumbtip with nail

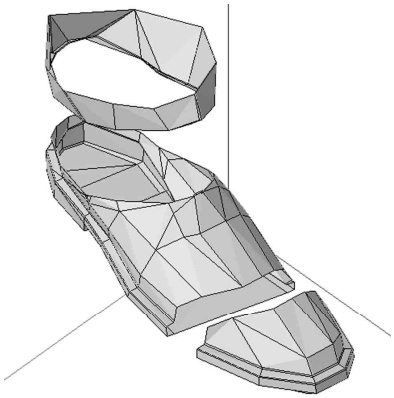
Partially deconstructed thumbtip with some relinked vertices (rear facets hidden)

Completed "gloved" thumbtip

Figures 19-21: Booting Up



Original Poser 3 male nude foot and toe parts, with ankle facets detached from foot.



Original Poser 2 male casual foot with toe detached; by discarding the cuff facets and joining the remains to the nude ankle above, we create a boot.

9. If the Shoe Fits ...

Now we come to the biggest task of this section: replacing My Hero's bare feet with a pair of shoes. The shoes will come from Poser 3's **Male Casual** model. Specifically, we're going to mate the ankle portion of the Nude Male foot to the shoe portion of the Casual foot. See **Figures 19-21**. Because they'll join seamlessly, our model will appear to be wearing close fitting boots of indeterminate length. (In the next tutorial we'll see how to give these boots a determined length.)

The four parts to be edited are named `rFoot, lFoot, rToe` and `lToe`.

Launch Ray Dream and open the geometry file `Geometries:newMaleNudeHi:maleNudeWithEyes.obj`. Select the objects `rFoot` and `rToe` and group them. With the new group still selected, select **Export** from the **File** menu and save the group as a Wavefront OBJ file. Name it `rFoot.obj`. Close `maleNudeWithEyes.obj`.

Open the geometry file `Geometries:maleCasHi:maleCasHi.obj`. Select the object `rFoot` (Poser 2 figures don't have toe joints.) and name it `rBoot.obj`. Close `maleCasHi.obj`.

Open `rFoot.obj`. In the **OBJ Import** dialog, make sure **One** is selected for the option **Create Meshform Objects**. This step is unique to Ray Dream because its native modelling environment does not handle vertex-level editing, and its Meshform Modeller, which does, is a separate environment. The Mesh Form Modeller, however, does not allow us to edit more than one part at a time. Importing the model parts as one object will allow us to work on both parts together in the Mesh Form Modeller. This step should not be necessary in any program that provides vertex-level editing natively. To be absolutely clear, the real goal here is to open the two parts in your vertex editor.

Refer to **Figure 19**. Select the facets making up the ankle portion of `rFoot` and detach them. Invert the selection and delete the remaining portions of the foot and the toe. Jump out to RayDream's native environment. Save the file but do not close it.

Open `rBoot.obj` as we did with `rFoot.obj`, making sure **One** is selected for the option **Create Meshform Objects**. Selects the facets making up the pants cuff portion of the foot. Pay particular attention to the very small and very thin facets on the underside of the cuff; these have to be selected as well. Since the plan is to get rid of the cuff, we needn't worry about selecting all the facets at once—we can simply select a few at a time and delete these as we go. Refer to **Figure 20**.

We also need to select the facets at the shoe's front and detach them in order to create a separate toe object. Again, refer to **Figure 20**. With the detached toe facets as the active selection, select Name Polygons from the Selection menu and call it `rToe`. Then invert the selection and name the remaining polygons `rFoot`. Jump out to RayDream's native environment. Copy the shoe to memory.

Let's return to `rFoot.obj`, which we left as a ring of ankle facets. Paste the shoe from memory into this document. Carefully align the shoe object with the ankle facets as closely as possible, being careful <u>not</u> to move the ankle facets in the slightest. When done, group the ankle and the shoe, and export the selection as `rFoot2.obj`. Save all files and close them.

Open `rFoot2.obj`, making sure **One** is selected for the option **Create Meshform Objects**. All three parts, the ankle facets, the shoe and the toe should be one object. Select the object (In Ray Dream you can select any contiguous group of polygons by double-clicking on any of its members.) and **Duplicate** it. Name one object `rFoot` and its copy `rToe`.

Jump into `rToe` to launch the **Meshform Modeller**. The Modeller will ask you if you wish to create a new master for the object; select **Yes** (the default value) and name the new master `rToe`. Once in the Modeller itself, select the toe facets and then invert the selection. With the ankle and shoe now selected, delete them and exit the Modeller.

Jump into `rFoot`. Select the toe facets and delete them. At this point we now have again separate foot and toe objects; the only task remaining now is uniting the ankle and shoe polygons of the foot object into a seamless whole.

We plan to do this by eliminating the edges of the shoe that meet the ankle and recontructing them using the actual vertices along the edges of the ankle. This technique can also be adapted to produce seamless edges between separate joints, as

shown in Figures 18-20.

The technique is very simple but can be very time consuming for large and/or complex edges. We'll be proceeding only one or two polygons at a time. Select a vertex on the edge of the shoe where it meets the ankle; delete it; create one or more new line segments between shoe and ankle by linking vertices that had formerly been linked to the deleted vertex to corresponding vertices on the ankle edge. Again, refer to Figures 18-20. Work your way around the edge, linking vertices and filling polygons until complete. Incidentally, it doesn't really matter whether we delete the ankle edge and link to the shoe or delete the shoe edge and link to the ankle; just choose one and be consistent.

We might also try using the Weld command to accomplish this without deleting points, but this command is only useful when we have two edges whose vertices are perfectly matched in number and generally matched in placement. This is not the case with our ankle and shoe—more vertices make up the edge of the ankle than the shoe. So new polygons must be created in order to build the row of facets that will unite the two pieces.

When complete, select the entire meshform, which is now one contiguous group of facets, and name it `rFoot`. Jump out of the Meshform Modeller and save the file, but do not close it. Select both `rFoot` and `rToe`, group them and copy them to memory.

Open `myHero.obj`. Delete `rFoot` and `rToe`. Paste `rFoot` and `rToe` from memory into the document and ungroup them. Check the names and if necessary, rename the parts; it is important that the object and master names of these parts be both correct and the same. Save `myHero.obj`.

To complete the left foot, first delete `lFoot` and `lToe`. Select `rFoot` and `rToe` and group them. Select Duplicate with Symmetry from the Edit menu. Ungroup `rFoot` and `rToe`. Move and orient the copies into place at the end of the left shin. Make the alignment as precise as possible without disturbing the left shin. Save `myHero.obj`.

Group `lShin` with `lFoot` and `lToe`. Export the group as `lFoot.obj`. Open `lFoot.obj`, making sure One is selected for the option Create Meshform Objects. When open, duplicate the object, naming one `lFoot` and its copy `lToe`.

Jump into `lToe` to launch the Meshform Modeller. The Modeller will ask you if you wish to create a new master for the object; select yes (the default value) and name the new master `lToe`. In the Modeller, select the toe facets. From the Polymesh menu select Reverse Polygon Normals. (Duplicating with symmetry in Ray Dream actually reverses the normals on mesh objects.) Select Invert from the Selection menu. With the shin and foot now selected, delete them and exit the Modeller.

Jump into `lFoot`. Select the toe facets and delete them. Select everything else and Reverse Polygon Normals. At this point we now have again separate foot and toe objects; the only task remaining then is creating a seamless edge for the foot to match the shin.

Referring to the technique described by Figures 18-20, delete all of the shin mesh except for the edge meeting the foot. Select a vertex on the edge of the foot where it meets the shin edge; delete it; create one or more new line segments between foot and shin edge by linking vertices that had formerly been linked to the deleted vertex to corresponding vertices on the shin edge. Work your way around the edge, linking vertices and filling polygons until complete. When done, select all and name the polygons `lFoot`. Jump out. Select `lFoot` and `lToe`, group them and copy them to memory.

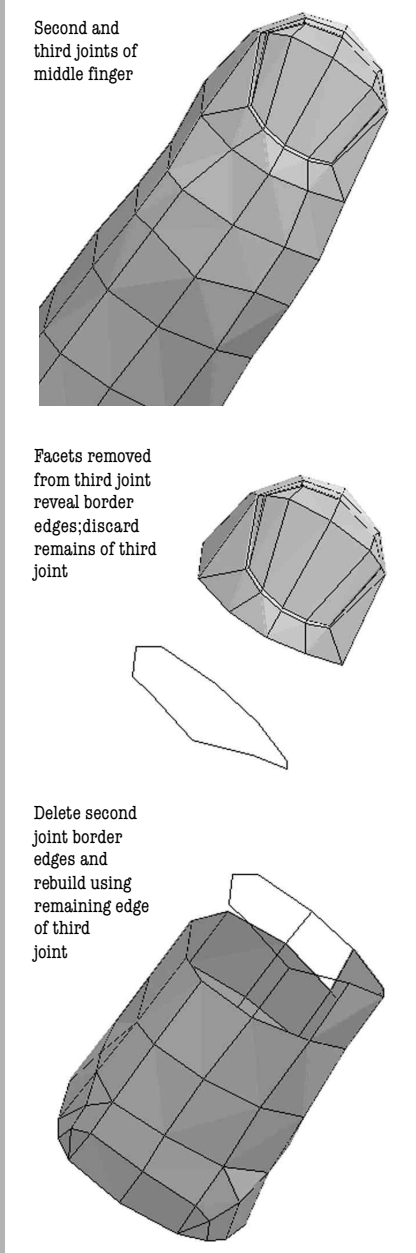
Open `myHero.obj`. Delete `lFoot` and `lToe`. Paste `lFoot` and `lToe` from memory into the document and ungroup them. Check the names and if necessary, rename the parts; it is important that the object and master names of these parts be both correct and the same. Save `myHero.obj`. We're done! Quit Ray Dream and launch Poser. Load My Hero from the New Figures library to check our work. At this point the most likely problems concern the normals on the left foot and toe; we'll know immediately if they're correct. If not, reopen `myHero.obj`, reverse normals as necessary and resave.

Congratulations! Our superhero base model is complete. Now go out and get a life, at least for an evening. Our next task, creating a finished model, Bat-Guy, will be far more time-consuming, and you'll need your strength.

Here endeth the lesson

■

Figures 18-20: Seamless Joints



When complete, the new edge of the second joint will blend perfectly with the edge of the third joint,because their vertices will be indentical,sharing the same coordinates.

Other documents and resources in this series include:

Primer on New Models

Preparing a Base Model

Building Bat-Guy

Appendix A: OBJ File Format

Appendix B: Poser Resource File Format

Appendix C: Joint Parameters Tables

Appendix D: Spherical Falloff Zones

Base Model Geometry and Resource Files

Bat-Guy Geometry and Resource Files

Final Touches: Adding the Cape

Cape Geometry & Morph Targets