Object Standards, Standard Objects

In December 1949 Martin Heidegger gave a series of four lectures in the city of Bremen, then an isolated part of the American occupation zone following the Second World War. The event marked Heidegger's first speaking engagement following his removal from his Freiburg professorship by the denazification authorities in 1946, and his first public lecture since his foray into university administration and politics in the early 1930s. Titled *Insight Into That Which Is [*Einblick in das, was ist*]*,[1] the lectures mark the debut of a new direction in Heidegger's thought and introduce a number of major themes that would be explored in his later work.[2] Heidegger opened the Bremen lectures with a work simply titled "The Thing" which begins with a meditation on the collapsing of distance, enabled by modern technology. "Physical distance is dissolved by aircraft. The radio makes information instantly available that once went unknown. The formerly slow and mysterious growth of plants is laid bare through stop-action photography."[3] Yet Heidegger argues that

> despite all conquest of distances the nearness of things remains absent. What about nearness? How can we come to know its nature? Nearness, it seems, cannot be encountered directly. We succeed in reaching it rather by attending to what is near. Near to us are what we usually call things. But what is a thing?[4]

This question motivates the lecture, and indeed much of Heidegger's later thought. In

---

[1] Heidegger, Martin. trans. Andrew J. Mitchell. *Bremen and Freiburg Lectures: Insight Into That Which Is and Basic Principals of Thinking*. Bloomington, IN: Indiana University Press 1994.

[2] Most notably they introduce the concept of the "fourfold," which while related to "The Thing" is beyond the scope of this chapter.

[3] Harman, Graham. "Technology, Objects, and Things in Heidegger" *Cambridge Journal of Economics* 2009 doi: 10.1093/cje/bep021 p. 5.

[4] Heidegger, Martin. *Poetry, Language, Thought*. 164

true Heideggerian form he answers, “The thing things.”[5] That is, it exists beyond our awareness of it or use for it; it has a material world all its own. In order to illustrate his argument, Heidegger selects a particular object to exemplify the nearness of things: a jug. The jug is an interesting intermediary; at once a constructed, built thing, it is also simple and utilitarian, made of natural materials, and crafted by hand. While one may be tempted to identify the thingness of the jug as its function - that is, its ability to hold, store, and dispense liquid - these are simply the ways in which it is made present to us. Its thingness exists outside of such use, and resides in its “independent stance”[6] [*Selbststand*] thus differentiating it from objects [*Gegenständ*], which stands toward us in their Being.

What then, is a simulated thing? What is the “thingness” of the process known as simulation? Simulation lies at the very heart of computation, as the computer may be understood most simply as a machine built for the simulation of any process that may be described procedurally. It is a tool built for the simulation of other tools, a technical object *par excellence*. For Heidegger, technology is the conversion of all things into an accessible surface[7], into their undifferentiated utility for us, a conversion from things into objects.[8] But what then is the materiality of technical things? As computer graphics began to crystalize as a discipline in the early 1970s, it became clear that while the realistic display of objects and environments was a primary concern, this concern necessitated a

---

[5] Heidegger, Martin. *Ibid.* 172.
[6] Heidegger, Martin. *Ibid.* 164.
[7] Heidegger would later elaborate on his thinking concerning technology in “The Question Concerning Technology”, but these views are first outlined in “The Thing”. See: Martin Heidegger, "The Question Concerning Technology", from *Martin Heidegger: Basic Writings from "Being and Time" (1927) to "The Task of Thinking" (1964)*, rev. ed., edited by David Farrell Krell. Harper: San Francisco.
[8] In this way it departs radically from thingness, though an object can co-exist as both object and thing, just as it can be at varying moments both present-at-hand and ready-to-hand.

theory of the nature of objects themselves. In other words, in order to simulate an object one must first understand and define what an object is. Over the course of the decade, computer graphics standardized the process by which simulated objects are constructed and understood. Objects became geometries onto which processes may be applied, which may interact with one another in specified ways. This marks the beginning of a shift in computer science away from procedural mechanization and toward a dynamic field of interactive objects. That is, research into computer graphics marks the beginning of a concern in computer science for nature of objects themselves, that is, for questions of ontology.

This chapter offers a historical account of the means by which computer graphics developed a theory of the object. In doing so it identifies a broad set of interests that galvanized the field at the start of the 1970s, principally: constructing computational objects to be produced by mechanical means, digitizing objects from the physical world, and standardizing those qualities that comprise a simulated object. As the means through which objects became simulated was standardized, a number of standard objects forms for testing and calibration began to appear and circulate among researchers in the field. The most famous of these is known as the Utah Teapot, and it offers an exemplary case study through which we might trace the development of CGI technology as it began to radiate out from the University of Utah toward the end of the decade. Both analog and antithesis to Heidegger's jug, the teapot ultimately offers a glimpse into the materiality of simulated objects, with implications far beyond the field of computer graphics.

From Lines to Curves

By the early 1970s computer graphics was beginning to move away from the challenge of simple object simulation and display. But while the problem of how to display wireframe models with hidden lines removed had been solved to a reasonable degree of accuracy, the shapes and forms available for simulation were severely limited. Simple Platonic solids were relatively easy to calculate, and a large number of objects could be made through the combination of these basic geometric forms. But to simulate objects from the real world, a much wider range of shapes with greater complexity would be necessary. The key challenge in rendering such objects was in accurately displaying irregular curves, as the introduction of curved surfaces meant an dramatic increase in the complexity of the object as understood by existing hidden surface algorithms. Some existing systems were able to remove hidden parts for curved surfaces,[9] but were only able to do so by restricting the class of possible surfaces, or by accepting a long execution time.[10]

Curved surfaces complicate the hidden surface problem in multiple ways. Perhaps most clearly, irregular curved shapes are the most likely to confuse the viewer if hidden surfaces are not removed, as the shape of a given object is not readily predictable and can easily overlap multiple times from a given perspective [FIG. 1]. Existing hidden surface

---

[9] These include: MAGI, Mathematical Applications Group Inc. "3-D Simulated Graphics," *Datamation*, vol. 14 Feb. 1968, p. 69; P.G. Comba, "A Procedure for Deflecting Intersections of Three Dimensional Objects," IBM New York Scientific Center, New York, NY Rep. 39.020, Jan. 1967; and R. A. Weiss, "Be Vision, A Package of IBM 7090 Fortran Programs to Draw Orthographic Views of Combinations of Planes and Quadratic Surfaces," J. Ass. Comput. Mach., vol. 13, Apr. 1966, pp. 194-204.

[10] Gouraud, Henri. *Computer Display of Curved Surfaces*. University of Utah Doctoral Dissertation. June 1971. p. 4.

algorithms functioned by applying a perspective transformation onto a given scene in which the viewer is located at infinity such that any given part of an object may be compared with another along the z-axis. In the interest of processing time each algorithm was designed to limit the number of comparisons it had to make to achieve a workable solution, and when the objects in a given scene are linear a program is only required to make one comparison for a given surface if it lies parallel to the XZ plane. If one were to apply a similar technique to a scene containing curved surfaces, they would have to compare segments of curves instead of line segments and the complexity of the problem to be solved would grow approximately as the square of the complexity of the surfaces to be dealt with.[11] In the very earliest models, "curved surfaces [were] portrayed by a set of privileged curves drawn on the surface. […] Each curve [was] in turn approximated by a succession of line segments."[12] This process created a polygon mesh that gave the approximation of a smooth curve [FIG. 2]. While these multi-faceted models could serve as functional approximations of a curved surface, they required a tradeoff between either a jagged, blocky appearance or a complex polygon mesh that made calculation difficult.

Thus what at first may seem like a relatively simple property of objects - curvature - in fact poses a number of significant challenges. These challenges are by no means unique to computer graphics, and in fact the study of irregular curvature holds a deep history within the field of mathematics and design that connects it to a number of technical practices developed in the second half of the twentieth century, most especially vehicle design and typography. Take, for example, the construction of a ship. The shape and curve of the ship's hull is an essential component of its construction, and the shape

[11] Gouraud, Henri. *Ibid*. 6.
[12] Gouraud, Henri. *Ibid.* 4.

must be both precise and symmetrical if the ship is to properly cut through water. Prior to the 1940s[13] designers who needed to employ such a curved surface would draft the line by hand; but in order to transfer that curve from a drawn surface to the ship or model a particular set of tools were needed.[14] Craftsman would use a long thin strip of wood known as a spline and bend it to the shape of the desired curve, using the drawing as a guide.[15] The wood would be held in place via the tension of the rod, and the shape would be maintained through a number of lead weights known as spline ducks[16], which push or pull on the spline at precise points to create the desired curve [FIGS. 3, 4]. Over time the wood would come take the shape of the curve of minimum strain energy and the weights could be removed, after which the spline could be used as a guide to transfer the shape of the curve onto the object itself.[17] The entire process is an example of a technique known

[13] In fact this history goes back much farther, to at least AD Roman times when ships were built using rib templates that could be reused over and over again. Drawn templates did not become popular until the 17th century in England, when the classical "spline" was likely invented and named. The earliest available mention of a "spline" appears in H.L. Duhamel du Monceau's 1772 treatise, *Eléments de l'Architecture Navale ou Traité Pratique de la Construction des Vaissaux*. For a detailed history see G. Farin, J. Hoschek, and M.-S. Kim. *Handbook of Computer Aided Geometric Design*. Amsterdam: North Holland Publishing Company (2002), Ch. 1.

[14] For a brief but thorough history, see: Farin, Gerald. "A History of Curves and Surfaces in CAGD" in *Handbook of Computer Aided Geometric Design*. G. Farin, J. Hoschek, and M.-S. Kim, eds. Amsterdam, The Netherlands: Elsevier Science B.V. (2002) 1-21.

[15] A contemporary tool that was used for smaller scale drawings was the French curve, which consisted of a number of templates made out of metal, wood, or plastic composed of numerous different curves. The tools came in a wide variety of shapes and forms, but the Burmester set of three curves are the most common. Many of the curves resemble the shape of a pistol, and the French name for the devices is indeed le pistolet.

[16] Spline ducks are so-named for their distinct shape, which resembles the head of a duck. Other names used by Schoenberg (1946) include "dogs" or "rats."

[17] Spline devices also helped bend the wood for pianos, violins, and other string instruments.

as “lofting,” in which a curve is transferred from a "lines plan" to a "full sized plan" in the construction of a streamlined object.[18]

Beginning in the 1940s a number of mathematicians began working to understand the function of the curves created by mechanical splines, with the first academic reference to splines appearing in Isaac Jacob Schoenberg’s “Contributions to the problem of approximation of equidistant data by analytic functions” in 1946[19]. This work culminated in a number of publications in the late 1950s and early 1960s from researchers at General Motors,[20] Boeing[21], and the British Aircraft Corporation.[22] Once splines had been introduced into the process of lofting and mechanical design they became introduced in a number of other contexts, most significantly in the design of motor vehicles at the French automakers Citroen[23] and Renault[24]. While the mathematical theory and industrial technique behind spline curves had been in development for decades and was first described by a number of other researchers, it was the work of Pierre Bézier at Renault that popularized their use in computer aided design and led to their adoption

[18] Lofting a ship requires a great deal of space, as the curves are transferred in a one-to-one ratio from the paper drawing to the spline and then onto the ship itself. Loft then refers to the space in which this practice took place, in the upper chamber or attic of a building. See: Forrest, A. Robin “Foreward” in *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling*. (vii)

[19] Schoenberg, Contributions to the problem of approximation of equidistant data by analytic functions, *Quart. Appl. Math*., vol. 4, pp. 45–99 and 112–141, 1946.

[20] C. de Boor. On calculating with B-splines. *J Approx. Theory,* 6(l):50-62, 1972.

[21] J. Ferguson. Multivariable curve interpolation. *J ACM*, ll(2):221-228, 1964.

[22] M. Sabin. Offset parametric surfaces. Technical Report VTO/MS/149, British Aircraft Corporation, 1968; M. Sabin. Parametric splines in tension. Technical Report VTO/MS/160, British Aircraft Corporation, 1970; and M. Sabin. Trinomial basis functions for interpolation in triangular regions (Bezier triangles). Technical report, British Aircraft Corporation, 1971.

[23] P. de Casteljau. Courbes et surfaces à pôles. Technical report, A. Citroën, Paris, 1963.

[24] P. Bezier. Definition numèrique des courbes et surfaces I. *Automatisme,* XI:625-632, 1966; and P. Bezier. Definition numèrique des courbes et surfaces II. *Automatisme,* XII:17-21, 1967.

by 3D computer graphics. Bézier curves, as they have come to be known, are used to model smooth vector curves that can be scaled indefinitely. The curves are produced by a set of control points which can be manipulated to transform the shape of the curve, and which function somewhat intuitively like spline ducks to pull a curve in a desired direction [FIG. 4]. Bézier put these theories into practice in 1968 with the development of a computerized system for the design and manufacture of cars at Renault called UNISURF,[25] which allowed designers to produce models in miniature which could then be divided into patches, reproduced through numerical control, and assembled into functional prototypes [FIG. 5]. Unlike earlier CAD systems like the DAC-1 - discussed in the previous chapter - the UNISURF system was able to not only assist in the design of motor vehicles, but also largely automate the process of prototype production through numerical control, making it one of the earliest CAD/CAM systems. Bézier's work on splines was further generalized in the 1960s into what are known as b-splines or basis splines, and NURBS, or non-uniform rational basis splines, which allow for more precise, individual manipulation of each curve along a given path.[26]

Unlike wireframe models in which curvature is approximated via a set of line segments, Bézier curves produce curvature through the simulation of natural processes,

---

[25] Bézier, Pierre. "Example of an Existing System in the Motor Industry: The Unisurf System" in *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, Vol. 321, No. 1545, A Discussion on Computer Aids in Mechanical Engineering Design and Manufacture (Feb. 9, 1971), pp. 207-218.

[26] While I will continue to use the term "Bézier curve" moving forward to describe a set of mathematical processes to approximate natural curvature, I am describing processes which are also largely true of b-splines and NURBS. I do not wish to gloss over important differences between these methods for reproducing curvature, though they each emerge from the same historical tradition described here. While their mathematical distinction is important, it is beyond the scope of this particular history. It should be noted, however, that all three curves are often present as options in a given software platform.

that is, through the mathematical approximation of multiple forces on a line. In so doing they accurately simulate the bounding surfaces of objects, which form the point of contact between surface forms. Because of their flexibility and scalability, Bézier curves have come to be used in most all CAD programs, as well as in programs that rely on smooth vector graphics such as Adobe Illustrator. The curves have also transformed the production of fonts and typefaces, with modern imaging systems like Postscript using cubic Bézier curves to draw the curved shapes and lines that make up letter forms[27] [FIG. 6].

From Curves to Surfaces

For 3D computer graphics, however, simple curves are not enough. While curves form the bounding edge of a given object, to model a realistic three-dimensional object a curve had to be interpolated into a surface; that is, from two- to three-dimensions. Research into 3D computer aided design and curved surface approximation began to take shape in the early 1970s, though its origins are much the same as those of Bézier and b-spline curves. The key difference here is the significance of interpolation to the process of transforming dimensions. While smooth lines can be approximated and formed from a relatively small number of data points, it becomes necessary when moving to surfaces in three dimensions to make certain assumptions about the consistency of a given object and interpolate its shape between a limited number of data points. While it is of course possible to construct objects using a massive number of data points that recreate its

[27] For more on this history, see: Knuth, Donald E. "Mathematical Typography" in *Bulletin (New Series) of the American Mathematical Society.* Volume 1, Number 2, March 1979.

surface with great accuracy, such a procedure is arguably impractical and, in the 1970s, computationally quite difficult. As such it was necessary to interpolate between lines and data points to form surfaces that may be build into structures. The key figure in this history is Steven Coons, who many consider the "grandfather" of computer graphics for his pioneering work on computer aided design at MIT in the early 1960s.[28] Coons' interest in CAD began like many of his contemporaries during World War II, when he was tasked with working on the design of aircraft surfaces. Much like Schoenberg, this experience seems to be the origin of Coons' thinking on geometric surface design, and by the early 1960s Coons was a researcher at MIT working in the Mechanical Engineering Department's Design and Graphics Group.[29]

While at MIT Coons served on the dissertation committees of both Ivan Sutherland[30] and Lawrence Roberts, both of whom made fundamental contributions to the field of computer graphics.[31] As was discussed in the previous chapter on hidden surface removal, this early research at MIT on interactive graphics is some of the earliest work to

---

[28] The most prestigious award granted in the field of computer graphics is the ACM SIGGRAPH's Steven A. Coons Award, given in odd-numbered years to an individual to honor that person's lifetime contribution to computer graphics and interactive techniques. The award was founded in 1983, three years after Coons death, and at the time of writing six of the fifteen recipients were at one time Utah faculty or students.

[29] One year prior in 1959 the Mechanical Engineering Department and Electronic Systems Laboratory of the Electrical Engineering Department at MIT had entered into a joint project sponsored by the US Air Force to explore the possibilities of a field they named "computer-aided design." These initial meetings are outlined in Coons, Steven Anson. "An Outline of the Requirements for a Computer-Aided Design System" *Proceedings of the AFIPS Spring Joint Computer Conference, 1963.*

[30] Ivan Sutherland's dissertation chair was Claude Shannon, with the third member being Marvin Minsky, a pioneer in the field of artificial intelligence.

[31] Coons also advised the dissertation research of Nicholas Negroponte, which also engaged the nascent field of computer aided design. See: http://archives.obs-us.com/obs/english/books/nn/bd1101bn.htm

engage the shift from two to three dimensions. While Sutherland's original Sketchpad program was largely built for the purpose of line drawings and drafting, subsequent versions elaborated on by Coons and Roberts introduced interactive 3D objects, culminating in Timothy Johnson's MS thesis *Sketchpad III: A Computer Program for Drawing in Three Dimensions* in 1963.[32] These later iterations of Sketchpad began to move away from the draftsman style of the DAC-1 and other contemporary systems, and began to resemble systems for object modeling that would form the foundations of the 3D graphics industry some ten to twenty years later.[33] Thus while Bézier and others were working on curves for CAD applications, Coons was thinking through surfaces for object construction, and in 1964 he developed what has come to be known as the "Coons Patch" for constructing and connecting interpolated surfaces, publishing his findings in 1967 in a text that would come to be known as "The Little Red Book" for its massive influence on the emerging field of geometric design.[34] Titled "Surfaces for Computer-Aided Design of Space Forms," the report presented the notation, mathematical foundation, and intuitive interpretation of a concept that would ultimately become the basis for a number of contemporary methods for surface description, including b-spline surfaces and NURB surfaces. In order to construct an object surface, Coons' technique pieced together a collection of adjacent patches with a set of continuity constraints that allowed the surface

[32] Johnson, Timothy. "Sketchpad III: A Computer Program for Drawing in Three Dimensions." in *Proceedings: Spring Joint Computer Conference, 1963.* 347-353.

[33] This distinction between drafting and CAD and 3D modeling and rendering - that is between the industrial manufacturing and visual entertainment aspects of CGI - is a long-standing one that reaches back as far as the 1960s. Once computer aided geometric design (CAGD) becomes formalized as a discipline at in 1974, the two fields develop in parallel but distinct from one another.

[34] Coons, S. A. "Surfaces for computer-aided design of space forms," M.I.T., Cambridge, Mass., Project MAC, Rep. MAC-TR-41, June 1967.

to hold the curvature expected by the designer. Each patch was designed from a set of four boundary curves - one on each side - and a set of blending functions that defined how the space between each curve was constructed through the interpolated values of the boundaries. That is, beginning with a set of four connected Bézier-style curves, one could construct a surface patch by interpolating the space between them, based on the shape of each individual curve.

For Coons this research was intended to facilitate the design of sculptured parts by designers with little or no experience in computing. The goal was to allow for a designer to quickly and simply produce a virtual prototype that could then be transformed to match the desired shape and complexity.

> [S]uppose a designer wishes to design an airplane fuselage, using the SKETCHPAD system. He would like to be able to draw the outline of the airplane as seen from the side, the outline of the airplane as seen from above, and some arbitrarily selected section midships. With these three arbitrary curves designed, he would like to have the computer automatically and immediately generate a "fair" surface and display this surface to him in sufficient detail so that he could make appropriate judgments. If the surface so generated does not satisfy him, he would perhaps like to modify his original design curves, or else he might perhaps like to add other new sections and have the computer automatically and instantly re-fair the surface to fit this additional information.[35]

As such it is a system built not to digitize existing physical objects, but to facilitate and replace the process by which objects are designed and produced.[36] Nonetheless, it does so through the digitization of existing methods for the design of shapes which heretofore

[35] Coons, S. A. *Ibid*. 2.

[36] Of course, a great number of the objects with which we interact on a daily basis are designed objects. As such, rather than digitizing the physical world through graphical techniques, this marks the beginning of the means by which objects are produced by computational means, and are subject to the constraints, limitations, and particularities of a computational system for design and production.

presented a great challenge to computing: complex curves and surfaces.[37] Significant here is the prevalence of a largely automated mathematical interpolation for the description of surfaces. No longer hand-crafted or even hand-calculated, objects are shaped by inferences drawn from edges that bound them, producing a particular aesthetic form. While the designer is free to modify the constraints of an object to increase its complexity, this interpolated form is nonetheless significant in defining the default shape of digital objects derived from this algorithmic tradition. The shape of digital objects interpolated in this way is no surprise when we consider the types of objects the Coons Patch was designed to build. As Coons notes in the abstract to his report, it is "[t]he design of airplanes, ships, automobiles, and so-called 'sculptured parts' [that] involves the design, delineation, and mathematical description of bounding surfaces."[38] Given the military funding of this research,[39] along with the inspiration for this work in the design of military aircraft during the Second World War, it is no surprise that the earliest digital objects are vehicles designed for slipstream movement.

By the early 1970s Coons has moved from MIT to Syracuse University where he served as faculty in the department of computer science. While at Syracuse he continued

---

[37] It should be noted that the display of curvature is in fact one of the earliest forms of computational visualization, though in a dramatically different form. Some of the earliest experimentation with computer visualization was done through the manipulation of early oscilloscopes, which cold be used to produce and transform simple geometric wave forms (sine waves, etc.) as well as read and visualize analog signal voltages. In some ways this transformation is indicative of a larger shift from analog to digital forms of visualization, in which continuous and irregular shapes such as curves become difficult to describe computationally.

[38] S. A. Coons, *Ibid.* iii.

[39] Coons' report was produced as part of the Project on Mathematics and Computation (Project MAC), which was funded largely by the defense department and responsible for several groundbreaking innovations in operating systems, artificial intelligence, and the theory of computation.

work in computer aided design, advising several foundational dissertations that would help generate this new field.[40] Key among them were those of Richard Riesenfeld and Elaine Cohen, who defended their PhDs from in 1973 and 1974 before becoming faculty at the University of Utah. Their research was inspired in part by a trip to paris in 1970 where Cohen and Riesenfeld met with Pierre Bézier to learn more about the ideas behind Bézier's new methods for CAD/CAM.[41] Riesenfeld's subsequent dissertation, titled Applications of B-Spline Approximations to Geometric Problems of Computer-Aided Design, introduced the concept of b-splines to the field of CAD/CAM. Then in 1974, shortly after arriving at Utah and with Cohen's support, Riesenfeld co-organized the first International Conference on Computer Aided Geometric Design (CAGD), bringing together several international pioneers in the field for the first time. Pierre Bézier and Steve Coons were featured speakers at the conference, which marked the first time both participated in the same technical event. The conference is credited as the origin of the field of CAGD, and its proceedings were subsequently published in a widely influential volume titled Computer Aided Geometric Design.[42]

It is here that a split in early graphics research becomes clearly defined. Much of the research conducted in the 1960s serves the dual purpose of both construction and

---

[40] One of the most influential texts to be written by one of Coons' advisees at Syracuse is Kenneth Versprille's *Computer-Aided Design Applications of the Rational B-Spline Approximation Forms*, which is the first published description of non-uniform rational b-splines (NURBS), which are widely used in modern CAD software.

[41] Author Unknown. "Richard Riesenfeld and Elaine Cohen: The 2009 Pierre Bézier Award Recipients" Solidmodeling.org: The Website for the Solid Modeling Association. Web. <http://www.solidmodeling.org/bezier_award/richard%20_riesenfeld_and_elaine_cohen.htm>. Accessed Sept. 13, 2013.

[42] *Computer aided geometric design: proceedings of a conference held at the University of Utah, Salt Lake City, Utah, March 18-21, 1974*. ed. Robert E. Barnhill. San Diego, CA: Academic Press [Harcourt Brace Javonovich] 1974.

simulation. That is, the issues that most interested the field were productive for both researchers who hoped to find new and more efficient ways of building material objects, and for researchers interested in finding more realistic ways to simulate material objects on the computer screen. The hidden surface problem is once again exemplary here. While on the one hand it is clearly a problem of simulation and the accurate display of non-transparent objects to the human eye, it is just as much a problem of design in which seeing all sides of an object at once is both difficult and confusing for a researcher or designer. As such it was of interest to a variety of institutions in the then-nascent field to solve the hidden surface problem in any number of ways. But by the early 1970s these fields begin to diverge, with CAGD crystalizing into a distinct field in 1974 and the simulation industry growing to demand interactive and rendered visual models for use in training, advertising, and other fields.[43]

Shading

For CAGD simple hidden line removal was enough for the accurate display of most objects, as the structure of an object was far more important than its realistic appearance. But for researchers concerned with rendering and the realistic display of solid objects the key problem to be solved was the accurate shading of complex surfaces. In the early 1970s there were several existing models available that allowed for object

[43] It is this second application that tends to receive the most attention when discussing computer graphics, as it the most visible and popular form of graphics, having transitioned from the military to the entertainment industry to mass appeal in the late 1980s and early 1990s. CAD/CAM does not capture public imagination in quite the same way as rendered images and computer animation, but it remains a widely influential field that seems set to grow significantly with the widespread adoption of 3D printing technologies that function largely based on CAD geometries.

shading with varying degrees of realism. A General Electric system used fixed colors assigned by hardware to each of the different polygons composing an object within a scene[44], which gave a colorful, cartoon-like appearance to an image[45]. Arthur Appel's ray casting system shaded surfaces as a function of the orientation of each polygon, which gave a more accurate greyscale appearance but was limited to simple geometric shapes [FIG. 7].[46] John Warnock's algorithm for hidden surface removal also allowed for surface shading based on both the orientation of an object and its distance from an observer [FIG. 8],[47] but all three programs were reliant on flat shading in which only one tone could be assigned to each face of an object.

> The essence of shaded pictures is to generate a different shade of grey for each resolution point on the projection screen. […] The requirement that the objects be composed of planar polygons was mainly made to facilitate the hidden-parts computations, but it also permitted simplicity in the computation of the shading for each polygon because a part of this computation is done in common for all the points of this polygon.[48]

For realistic images with smooth, continuous surface shading that approximated objects in the physical world an additional calculation was required. Much as with Coons' early work on interpolation for two- to three-dimensional transformations, the simplest solution to the problem of smooth shading that preserved simple hidden surface removal techniques was to simply add another step to the process, in which the shade values of an object were calculated.

---

[44] R.S. Rougelotand and R.Shoemaker "G.E. real time display, "General Electric Co., Syracuse, N. Y., NASA Rep. NAS 9-3916.
[45] Gouraud, Henri. "Continuous Shading of Curved Surfaces" *IEEE Transactions on Computers*. June 1971. 623.
[46] Appel, Arthur. "Some Techniques for Shading Machine Renderings of Solids" *Sprint Joint Computer Conference, 1968.* 37-45.
[47] Warnock, John E. *A Hidden Surface Algorithm for Computer Generated Halftone Pictures*. University of Utah ARPA Report. RADC-TR-69-249. June 1969.
[48] Gouraud, Henri. "Continuous Shading" 624.

There are two key figures in the history of shading. Both were University of Utah graduate students, though both came from somewhere well outside Salt Lake City. During this period the university drew a large number of graduate students from France, particularly from the National Institute for Research in Computer Science and Control (INRIA) in Rocquencourt and from Ecole Polyytechnique in Palaiseau, both on the outskirts of Paris[49]. The first French such student in the department was Henri Gouraud, who had graduated from the École Centrale Paris in 1967. Gouraud's work at Utah was in the area of object rendering, specifically in shading algorithms used to create realistic light and shadow on a graphically rendered object. His dissertation offered a means for rendering and shading complex curved objects through linear interpolation across individual polygonal facets, allowing for the continuous shading of surfaces.[50] Prior to this most programs used the aforementioned method known as "flat shading," which calculated the shade of each polygon in its entirety and could not be used to compute a smooth, continuous surface between polygons. Gouraud solved this problem by developing an interpolation method which estimates the surface normal[51] of each vertex in a polygonal 3D model and makes lighting calculations based on a reflection model. Color and shading values can then be interpolated for each individual screen pixel based on the values calculated for these vertices. While Gouraud shading is more procedurally demanding than flat shading, it produces a much more realistic appearance [FIG. 9]. Significantly this method does not transform the geometry of the object itself, simply the

---

[49] Crow, Jim. Unpublished email interview with Peter Shirley, 8 Nov 1996.
[50] Gouraud, Henri. *Computer Display*.
[51] The "normal" of an object is a line that runs perpendicular to its surface. In a sense it is the inverse of a tangent line or plane to a given point on a surface, in that it forms a 90 degree angle with the tangent.

appearance of its surface. A sphere rendered through interpolated shading will still have visible flat edges along its outer surface, though the effect is minimized by the shading illusion. The more polygons used to construct the object's geometry, the smoother the surface will appear. Gouraud's final dissertation included multiple examples of the shading algorithm applied to early three dimensional models, most famously a model of his wife Sophie Gourard's face.[52] The face was digitized manually by drawing polygonal sections over half of Sophie's face, photographing the results, and building a model in the Utah computer laboratory. The face could then be copied and flipped to produce a symmetrical image that approximated a mask made of the face of Sylvie Gouraud [FIGS. 10, 11, 12]. Gouraud's viewed his method as a way of more accurately representing the curved geometries of the body. This would allow for a wider range of objects to be simulated, and also creates a second means of representing irregular curved surfaces. While Bézier curves are useful for the bounding surfaces of mechanical objects, it is difficult to render a complex curved shape like the human body with a single snaking curve. Polygon meshes allow for an object such as a face to be constructed and carved, and then transformed via interpolative shading algorithms into something that more closely approximates the human body.

Gouraud was not the only French student to work on the problem of shading. The same year Gouraud defended his dissertation, the department took on a new graduate student from the INRIA in Paris, a young Vietnamese man named Bui Tuong Phong whose family had fled to France during the Vietnam War.[53] While at Utah Phong picked

---

[52] Gouraud, Henri. *Computer Display*. 55.
[53] Phong was born in Hanoi, Vietnam in 1942 on the eve of the First Indochina War. While Vietnam had been a French colony since the mid-19th century, the ensuing

up much of the work Gouraud had completed with his dissertation, working on both shading and reflection algorithms for 3D rendered models. In his 1973 thesis he describes a shading model that overcomes several of the difficulties of Gouraud shading.[54] While Gouraud's model interpolates colors using a linear gradient derived from the vertex normals of each polygon, in Phong shading a separate color computation is made for each individual pixel. [FIG. 13] As one might imagine, this makes Phong shading much more computationally expensive than Gouraud shading, and as such it was considered highly experimental at the time of its development and was not widely used until fairly recently.

Along with this shading model, Phong also developed a reflection model based on his own observation that shiny surfaces have small intense specular highlights while dull surfaces have large highlights that fall off gradually. The model combines three forms of lighting simulation to produce the final effect: an ambient light that is spread evenly over the entire object, a diffuse reflection on an objects rough surfaces, and a specular reflection on an objects shiny surfaces. While Phong shading produced a more accurate model for object opacity, it is Phong reflection that is the more significant breakthrough, as it allows for the rendering of highlights before it was possible to create rendered reflections through ray tracing. This reflection model may be used in conjunction with

conflict between the Viet Minh and the French ultimately led to the dissolution of French Indochina on July 20, 1954. That same year Phong's family would move from Hanoi in the north to Saigon in the south, where he attended the Lycée Jean Jacques Rousseau. In 1964 he moved to France and was admitted to the Grenoble Institute of Technology, where he received his *Licence en Sciences* in 1966 and his *Diplôme d'Ingénieur* from the École nationale supérieure d'électronique, d'électrotechnique, d'informatique, d'hydraulique et des télécommunications (ENSEEIHT) in Toulouse in 1968. Following that he joined the INRIA as a researcher in Computer Science, working in the development of operating systems for digital computers before being recruited by Sutherland to come to Utah in 1971.

[54] Bui Tuong Phong, *Illumination of Computer-Generated Images*, Department of Computer Science, University of Utah, UTEC-CSs-73-129, July 1973.

Phong shading, or may be used separately with other shading models such as Gouraud shading. Given that computer hardware, software, and languages frequently change and are replaced over time, the endurance of these early models such as Gouraud and Phong shading may seem surprising. Each is still widely used as an industry standard, and in contemporary 3D computer graphics suites such as Autodesk Maya, Phong and Gouraud shading still operate as objects that may be applied to a given geometry.[55]

Digitizing the Physical

With functional solutions to hidden surface calculation, the ability to construct and modify irregular shapes through Bézier and b-spline curves, and realistic rendering algorithms for half-tone shading, an entire world of objects had become available to computer graphics. By the late-1970s the field would become deeply invested in what is often considered the “holy grail” for the graphics scientific community, that is the quest for ultimate visual realism, for perfect simulation.[56] This moment, then, marks the beginning of a slip between the digital and the physical - the moment in which computer graphics begins to lay claim to the realistic simulation of the physical world. The earliest work on object simulation was done for the purpose of assisting in the design of particular object forms, largely for the military and the transportation industry. Yet these new tools for design came to influence the means by which all manner of objects were produced, and in turn transformed the shape of “designed” objects of all sorts. But what of so-called “natural” objects which are not mathematically designed *a priori*? By the

[55] Shortly after graduating Phong accepted a job at Stanford University, though he had been aware for some time that he was diagnosed with terminal Leukemia and passed away in 1975.
[56] See Barbara Flückiger: Visual Effects. Filmbilder aus dem Computer, Marburg 2008.

early 1970s, researchers were engaged in efforts to produce new ways of capturing the physical dimensions of objects through automation. The impulse for this research is quite different than the earlier work of Schoenberg, Bézier, Coons, and the like, and it is indicative of the split in graphics research between those interested in computer aided geometric design (CAGD) for the purposes of industry, and those that were interested in rendering algorithms for the display of images. "In the early 1970s at the University of Utah, there was substantial activity in the development of rendering algorithms," Frank Crow recalls in a piece for the IEEE,

> there was a constant shortage of data for interesting shapes to be displayed with these algorithms. Tiring quickly of spheres, cubes, tubes, and other easily generated shapes, interesting efforts were often mounted to capture more elaborate data. One set of efforts took the form of developing automated methods for capturing physical measurements. 3D digitizers using mechanical means, photogrammetry, and even lasers were developed. However, much of the interesting data was completely handcrafted."[57]

This data was pulled, most often, from that which was near. Either sketched on the fly or traced and mapped directly from the surface of existing physical objects, a number of important experiments took place at Utah in the first half of the decade.

One of the earliest attempts at digitizing a physical object was done by graduate students in Ivan Sutherland's introductory computer graphics course in 1972. Robert McDermott, a former graduate student at Utah and current faculty in its School of Computing, recalled that

> Sutherland challenged his students to choose something iconic to realistically render. We selected the Volkswagen Beetle—as a symbol of global culture,

---

[57] Crow, Frank. "The Origins of the Teapot" *IEEE Computer Graphics and Animation.* Vol. 7, Iss. 1. Jan. 1987. 8.

> because it was large enough to measure as a group, and because Ivan's wife, Marsha, owned one.[58]

They used what is known as a "points and polygon" description, whereby the car was divided into parts such as the hood, roof, and trunk, and those parts were measured using a set of points and lines drawn directly on its surface.

> Jim Clark (PhD '72) and I were taller so we arranged to have the higher points of the car. Bui Tuong Phong (PhD '73) and Raphael Rom (PhD '75) were shorter so they measured the lower sections. […] A volleyball stanchion and joints in the pavement formed a three-dimensional reference system. We used yardsticks to measure the x, y, and z coordinates of the painted points on the car surface.[59] [FIGS. 14, 15]

As the car was a symmetrical object, the students only measured one of its halves, which could then be flipped and mirrored to create the vehicle in its entirety. The process was slow and tedious, and took several class sessions to complete. As such, Marsha Sutherland spent several weeks driving around Salt Lake City with the makings of a polygonal grid painted on one half of her car.

The car data was ultimately entered as a list of point coordinates into text data files, and rendered using a system developed by Gary Watkins (PhD '70) to imprint shaded images onto a direct film recorder, taking several minutes per photographic image.[60] Significantly the car was never intended to be constructed as a whole. As this was a class assignment each student had only measured and produced a small section of the car, the result of which was a number of images and geometric data of car panels and parts. When one of the computer lab staffers decided to try and put the pieces together,

---

[58] McDermott, Robert. "Robert Remembers: The VW Bug". *The Utah Teapot: Quarterly Newspaper for the Alumni and Friends of the University of Utah School of Computing.* Fall 2003. p. 7.
[59] Mcdermott, Robert. *Ibid*. 7.
[60] Mcdermott, Robert. *Ibid*. 7.

they found that small errors on the part of each student made for a highly irregular model, which took weeks to adjust into an identifiable object.[61] The final image was not without its share of problems. The model consisted of little more than a skin or surface, and lacks any depth or interior. What's more, it is missing several key features such as headlights and wheels. Still the image was used as a test object in several research papers and occasionally appears in contemporary journals and animated features [FIGS. 17, 18].

That same year Ed Catmull (PhD '74) had used a similar system to digitize his own hand for use in what is the earliest example of fully shaded half-tone 3D computer animation.[62] To make the model

> a plaster of paris mold was made. [...] Next, a plaster model was made from the mold and covered with a thin layer of latex material. Polygons were drawn on the latex. It was necessary to extract the three dimensional coordinates of some 270 corner points defining the surface of the hand, organize them into about 350 polygons, and organize the polygons into the parts of the hand.[63] [FIG. 19, 20]

Once the hand had been mapped it was digitized using a coordinate measuring machine or CMM, consisting of a table on which a large gantry-style rig was constructed with an articulated arm [FIG. 21]. The arm was fitted with a grasping device that, with the push of a foot pedal, captured the precise location of the object it was touching in threespace. By capturing each of the vertexes of the plaster model, Catmull was able to then digitally reconstruct the hand [FIG. 22]. After the hand was modeled it was animated using a motion picture programming language called MOP.[64] The resulting film[65] also included

---

[61] The technician, Dennis Ting (BS '71) also compared the human time cost vs. the computer time cost and found that the image of the VW cost more than its street value.

[62] Catmull, Edwin. "A System for Computer Generated Movies" *ACM '72 Proceedings of the ACM annual conference.* Volume 1. 1972. 422-431.

[63] Catmull, Edwin. *Ibid*. 429.

[64] MOP will be discussed in a later chapter as it relates to object-oriented systems design at Utah in the early 1970s.

the work of Fred Parke (PhD '74) who constructed a three dimensional face using a similar method to Gouraud, but with the addition of flexibility and movement whereby the face could smile, frown, and purse its lips using linear interpolation [FIG. 23].[66] Both Catmull's hand and one of Parke's faces would later be used in the film *Futureworld* (1976),[67] in what is considered the earliest example of 3D computer animation in a feature film.[68] [FIG. 24]

As these examples demonstrate, computer graphics in this moment finally began to produce the kinds of objects we now associate with contemporary digital images, and by the mid-1970s the field had moved beyond the kinds of structuring concerns of the previous ten years and toward a new interest in the accurate representation of specific real-world objects and interactions - that is, a broad but shallow mimesis. All of this, however, was dependent on two key developments. As Catmull describes it, it was only "[w]ith the recent developments in fast hidden surface algorithms and a method for smooth shading of half-tone pictures, [that] it has become feasible to generate useful movies with the computer."[69] The first of these is the work of Lawrence Roberts, John Warnock, Gary Watkins, and others - described in detail in a previous chapter - and the second is the work of Henri Gouraud and later, Bui Tuong Phong. Thus it is not until decades into the development of computer graphics that it becomes possible to realistically approximate three dimensional objects as they appear in the physical world.

---

65 Catmull, Ed and Fred Parke. *A Computer Animated Hand*. 1972. Film. <http://vimeo.com/16292363>

66 Parke, Fred. "Computer Generated Animation of Faces" *ACM '72 Proceedings of the ACM annual conference*. Volume 1. 1972. 451-457.

67 Heffron, Richard T. *Futureworld*. MGM (1976) DVD.

68 Miller, John. "Futureworld" *Turner Classic Movies*. Web. http://www.tcm.com/this-month/article/245820%7C0/Futureworld.html Accessed Sept. 14, 2013.

69 Catmull, Edwin. *Ibid*. 422.

While we often associate graphics with this playful illusion, an analysis of this early history reveals a much broader set of interests and influences that shape what a simulated object has come to be.

Standard Objects

Over the course of the 1960s and early 1970s, computer graphics standardized the process by which objects are produced, replicated, and digitized. Through the standardization of objects as geometric, mathematical models, the field created a base on which to build future research. No longer exclusively concerned with geometric construction, researchers could focus on specific objects and processes, improving on the realism of individual properties through simulation. Even today a great deal of research in the field is directed toward the simulation of a narrow set of particularly challenging effects – hair, skin, water, and other natural phenomena, for example. This emphasis on a small subset of parts in any given world produces what Lev Manovich has described as the highly uneven ontology of computer graphics, whereby realism is unequally distributed in a particular simulation.[70] Yet paradoxically, this inequality is predicated on a flattening of the object world. While only a small handful of processes are made available for simulation, that simulation may be enacted equally on any object. That is, while we may consider certain properties to "belong" to certain objects - the shattering of glass, the movement of fur, the translucence of skin, etc. - we may simulate these processes on any object, due to its standard form broadly resembling that of any other

[70] Manovich, Lev. "Assembling Reality: Myths of Computer Graphics" in: *Afterimage* 20/2 (1992), pp. 12–14, under http://www.manovich.net/TEXT/assembling.html.

objet - as a texture, a class, an algorithm. In other words, specific objects aren't necessary, specific properties are.

While all objects become alike in that they are subject to the same set of conditions for their construction, some objects have become particularly salient to the field. These objects are most frequently deployed in the testing of new graphical algorithms, and are often included in computer graphics research and products as a kind of nod to the history of graphics. They operate, in effect, as the standard objects of computer graphics, the objects onto which each new algorithm is applied and tested. What is unique in the case of graphical standards, is that the standard quality these objects contain is precisely "object-ness" itself. Of course media standards are by no means exclusive to computer graphics, in fact just about all media formats have a set of standard objects that are used to gauge the proper functionality of the technologies required for their reproduction.[71] Media standards are a by-product of mass production, and function as a kind of quality assurance mechanism, allowing reproductions to be tested against a given standard specific to that medium. Media standards are often the best place to look if one is interested in understanding the functionality of a given technology, since those objects chosen for standardization often bear the markers of the process of production, in that they are designed to make it readily apparent when something in that process goes wrong. How these objects come to be standardized is a matter of both cultural history and material specificity, and decoupling these objects from their long history of use can prove particularly challenging. The purpose of such

[71] Other examples of media standards include Susan Vega's single "Tom's Diner" which was infamously used to develop the audio compression scheme that would come to be known as the MP3. For more on the history of the MP3 see: Sterne, Jonathan *MP3: The Meaning of a Format*. Durham, NC: Duke University Press (2012).

standards is generally that they are unexceptional objects in that they conform to a large number of typical parameters, and as such their development is rarely heralded as a significant event. There is rarely an academic paper or research report to commemorate their creation, as they are tools to be used in the service of research and not often research in themselves. And so instead of archives we have anecdotes told in journals or at conferences years later, once the cultural significance of the standard has been realized. As Ann Sophie Lehmann notes in her work on one such standard, the everyday quality of this engagement is both challenging and essential to the study of these practices. As we will see below, it is often from the everyday world of the researcher that the objects of research are born, that is

> before there was code, there was direct engagement with analogue materials. This kind of engagement is difficult to study. If we are not there to witness it ourselves, we have to rely on visual or verbal descriptions, which are often anecdotic in nature, or strike us as such, because they are so straightforward. For many computer scientists research starts by finding the right objects and materials at home, outside or in DIY stores, by observing how they are visually and physically composed, and finding ways to record their make-up. Many texture, shading and illumination algorithms are preceded by such tinkering with the analogue.[72]

Given this, we might view the complexities and inconsistencies in the historical narrative of these objects as both challenges and assets to this history. How then does an everyday object become the placeholder for every possible object? What kind of ontological work does that operation perform, and what does it tell us about computer graphics as a medium and a practice?

---

[72] Lehmann, Ann Sophie "Taking the Lid off the Utah Teapot" *Zeitschrift für Medien- und Kulturforschung.* No. 1/12 "Entwerfen" 178-180.

Utah Teapot

In 1974, Martin Newell, a young researcher at the University of Utah, was looking for an object that would move 3D graphics from simple geometric shapes such as spheres and cubes into the realm of recognizable, real-world objects. At the time Newell was a graduate student completing his dissertation research, having emigrated to the US from England where he had worked for the CADCentre, a government-funded research institute for computer-aided design techniques.[73] While at home one day for afternoon tea, his wife Sandra suggested he model the teapot they had recently purchased[74], a German-made Melitta whose design dates back to the 1950s [FIG. 25]. The pot's simple, modern shape was an ideal fit for the then-current challenges to the field of computer graphics: it was a convex object, contained saddle-points, had a concave element (the hole in the handle), it self-shadowed, had hidden surface issues, and looked reasonable when displayed without a complex surface texture. It was complex enough to test the problems that challenged the field, but was also an elegant, simple, and recognizable design. Newell quickly took a rough sketch of the teapot's profile on graph paper, capturing the essence of its shape but not its precise dimensions [FIG. 26].[75] From here he guessed at the location of suitable control points for the cubic Bezier splines to form

---

[73] van Dam, Andries. “Some Personal Recollections on Graphics Standards” ACM SIGGRAPH Vol.32 No.1 February 1998. http://www.siggraph.org/publications/newsletter/v32n1/columns/carson.html

[74] The teapot was purchased at the Zions Cooperative Mercantile Institution or ZCMI. ZCMI is one of the earliest department stores in the US, founded in 1868 by Brigham Young.

[75] Crow, Frank “The Origins of the Teapot” 1987

the teapot's curved surface,[76] and then measured these points using the graph paper grid.[77] To turn this sketch into a three dimensional model, Newell treated the lid, rim, and body of the profile sketch as a surface of revolution that could be carried around 360 degrees to form a closed surface. He then extrapolated the control points for the handle and spout, and constructed them using three dimensional tubes. Once the data set had been generated, it was processed into a series of patch parameters that could be input into a line-drawing program using a Tektronix storage tube connected to a DEC PDP-10, in order to display the teapot as a wire frame.[78] The final patch array consisted of 144 numbers,[79] famously lacking any bottom geometry [FIG. 27, 28].[80]

The teapot was by no means immediately taken up as a standard, and its standardization was never Newell's intention in constructing the object to begin with. The teapot's first appearance in publication is in Newell's dissertation, defended in 1975, where it is one of many objects used to demonstrate his research into procedural modeling, including Sutherland, et al's 1971 VW Bug model, a number of chess pawns[81],

---

76 The teapot is one of the first 3D CG object that was rendered as sculptured surface with Bezier curves, rather than as a set of polygons.

77 Author Unknown. "The Utah Teapot" Computer History Museum. Web. http://www.computerhistory.org/revolution/computer-graphics-music-and-art/15/206 Accessed Sept. 15, 2013.

78 Baker, Steve. "The History of the Teapot" Web. http://sjbaker.org/wiki/index.php?title=The_History_of_The_Teapot

79 Before the model began to be included as a standard object in rendering software such as Autodesk's Maya, many researchers would memorize the numbers and could recite them in order.

80 While only the teapot has become famous as a standard object model, Newell in fact modeled his entire tea set including four sets of cups and saucers, four spoons, and a small milk jug. The cup and saucer sets are also missing their bottoms.

81 The pawn set and checkerboard floor are arguably the earliest iconic images in computer graphics, with examples dating back as far as 1968. Pawns are simple uniform shapes that can be easily modeled, and the grid of a chessboard is a useful means of demonstrating depth, perspective, and distortion of objects in a scene.

and a carousel complete with toy airplanes.[82] The teapot is never singled out as a unique object, and is shown only as part of a larger "table setting" and in a collection of "various objects defined using Bézier patches."[83] It isn't until the following year that the teapot began to gain popular recognition, when it is used in a paper co-authored by Newell and colleague Jim Blinn[84] to demonstrate and expand on the texture mapping technique first developed at Utah by Ed Catmull in 1974.[85] Titled "Texture and Reflection in Computer Generated Images," the paper was a huge leap forward in terms of visual realism, with over half a dozen permutations on the teapot with various textures, lighting, and reflections [FIG. 29]. Blinn saw the teapot as an ideal object for testing, as its surface textures could resemble ceramic glazes with varying degrees of reflectance, producing highly realistic and diverse images.[86]

Significantly, the shape of Blinn's teapot is different than that of Newell's. The original Melitta is slightly taller than the now famous Utah teapot, as are the original renderings that Newell made in 1974. This transformation seems to have been the work of Blinn, though there are several distinct explanations for the change, equally cultural, technical, and aesthetic. The first is that some time in 1976, Blinn slightly squashed the teapot to demonstrate the abilities of graphics in a funding application to the department of defense. "The joke – implying that defense would like simulations of squashing things

---

[82] Newell, Martin. *The Utilization of Procedure Models in Digital Image Synthesis*. University of Utah Doctoral Dissertation. Summer 1975. UTEC-CSc-76-218.
[83] Newell *Ibid.* 84-88.
[84] Blinn, James F. and Martin E. Newell. "Texture and Reflection in Computer Generated Images" *Communications of the ACM* Oct. 1976 Vol. 19 No. 10. 542-547.
[85] Catmuli, E.A. "Computer display of curved surfaces." *Proc. Conf. on Comptr. Graphics, Pattern Recognition, and Data Structure*, May 1975,pp. ll-17 (IEEE Cat. No. 75CH0981-1C).
[86] Crow, Frank. *Ibid*. 8.

– was too subtle and no funding was obtained, yet the change in form gave the teapot its characteristic, slightly more cartoonish appearance, which it was going to keep."[87] A second explanation is that Blinn was rendering the teapot on a framebuffer developed by Evans & Sutherland whose pixels were not square, but rectangular.[88] Thus in order to make the model conform to the pixel raster, Blinn squashed it rather than scale the image.[89] Still a third source produced from conversations with Blinn himself notes that "[t]he original data was scaled a number of times in its early life. Jim Blinn recalls that the current shape was felt to be the most esthetic at some point," and so it stuck.[90] Blinn would further popularize the teapot in a number of subsequent publications after leaving Utah to work at NASA's Jet Propulsion Laboratory. The data set would also follow Ed Catmull to the New York Institute of Technology, and later to Lucasfilm, where numerous test images would use its distinctive shape. Indeed one effect of the large network of influence that the Utah school had on the subsequent development of the field of computer graphics was the wide distribution of Newell's teapot through informal networks to a variety of research institutions.

Around 1986 Frank Crow - a former graduate student at Utah - posted the original teapot Bézier patch control points to the Net, making the teapot model much more

[87] Lehmann, Ann Sophie "Taking the Lid off the Utah Teapot" 173-174
[88] Blinn refutes this version of the story, suggesting the decision was simply an aesthetic choice made when playing around with the test object. See Seymour, Mike. "Founders Series: Industry Legend Jim Blinn" *FXGuide*. Web. http://www.fxguide.com/featured/founders-series-industry-legend-jim-blinn/ Accessed Sept. 15, 2013.
[89] Baker, Steve. *Ibid*.
[90] Crow, Frank. *Ibid*. 9.

broadly accessible.[91] As the data was free to use, researchers working on a new texture, lighting, or effects algorithm could quickly drop in the teapot as a test object, rather than go through the laborious process of designing and entering their own geometric data. Due to this broad accessibility along with the model's relative simplicity and effective geometry it was readily taken up by CGI researchers at institutions across the country, and has appeared in thousands of research papers, SIGGRAPH panels, and graphics demos over the past forty years. The teapot has become an icon, and is likely the most famous and widely distributed CGI image in the history of the medium.[92] It is so important to the field of computer graphics that it has in a sense become one of the base geometric objects - or *primitives* - from which all other objects are constructed. In one famous ray-traced image by Jim Arvo and Dave Kirk from their 1987 SIGGRAPH paper "Fast Ray Tracing by Ray Classification," six stone columns stand alone, each topped

---

[91] While the date of 1986 is somewhat vague and does not seem to be directly substantiated, at the latest the Bezier patches for the full teaset were posted by a Juhana Kouhia on October 25, 1991 ti NIC.FUNET.FI, an FTP server hosted via the Finnish University and Research Network (FUNET). See, http://www.cs.utah.edu/gdc/projects/alpha1/help/man/html/model_repo/model_teapot/model_teapot.html and http://www.sjbaker.org/teapot/teaset.tgz

[92] Since the digitization and standardization of the teapot, many other standard graphical objects have followed suit. Subsequent objects are generally more complex, or are produced using different methods that allow for unique forms of testing. The Stanford Bunny, developed by Greg Turk and Matt Levoy in 1994, consists of data describing 69,451 triangles determined by 3D scanning a ceramic figurine of a rabbit. The method for digitization is clearly different here. 3D scanning was in its infancy in the early 90s, and the bunny suffers from many of the problems inherent to this method of digitization, such as holes in its data that must be compensated for in any algorithm applied to it. The Stanford Dragon is a similar model developed in 1996 also using 3D scanning technology, with data describing 871,414 triangles. A third modern standard – a cartoonish orangutan face named "Suzanne" after the monkey in the 2001 Kevin Smith film Jay and Silent Bob Strike Back – is much closer to a modern Utah teapot. With only 500 faces it is an incredibly simple model that can be used to quickly test basic material, animation, rigs, texture, and lighting setups, and is also used as a joke image in demos and games.

with a geometric object.[93] On five of the columns sit the platonic solids (tetrahedron, cube, octahedron, dodecahedron, icosahedron) - those regular, convex polyhedra with congruent faces of regular polygons and the same number of faces meeting at each vertex - and atop the sixth sits the Utah teapot [FIG. 30]. The image is titled "The Six Platonic Solids," which has given the teapot the playful nickname “teapotahedron.” The teapot is included as a default in most contemporary rendering and geometry programs such as AutoCAD, Lightwave 3D, and 3ds Max. Some RenderMan-compliant renderers even support the teapot as a built-in geometry accessible simply by calling RiGeometry("teapot", RI_NULL).

As computer graphics have become more visible and accessible to everyday users, so too has the teapot. It appears as an “Easter egg” in a number of places, perhaps first as an object in the Windows operating system’s “Pipes” screensaver, released in 1994 alongside Windows NT 3.5 [FIG. 31].[94] In 1995 the teapot was featured prominently in Pixar’s first feature length film *Toy Story*[95] during a tea party scene [FIG. 32], and it also appears in an October 1995 episode of *The Simpsons*[96] which featured a brief scene rendered in 3D [FIG. 33]. The object functions as both a nod to the history of the medium, and a knowing wink to those researchers and artists who understand its broader

---

[93] Arvo, James and David Kirk. “Fast Ray Tracing by Ray Classification” *Proceeding: SIGGRAPH '87 Proceedings of the 14th annual conference on Computer graphics and interactive techniques*. ACM SIGGRAPH. 55-64.

[94] The screensaver has a very small chance of spawning the teapot, and there are numerous threads on the Web devoted to stories of teapot spottings and time wasted waiting for the teapot to appear.

[95] Lasseter, John. *Toy Story*. Disney/Pixar ; Burbank, CA : Distributed by Buena Vista Home Entertainment, Inc., [2005] DVD.

[96] Groening, Matt. “Treehouse of Horror VI” *The Simpsons.* New York, NY: Twentieth Century Fox. 1995.

significance. The teapot's appearance as an everyday object allows it to hide in plain sight, a CGI analog to what Erwin Panofsky has described as "disguised symbolism."[97] By tracing the teapot's transformation as it becomes diffused through both informal and formal networks, and is reproduced in countless iterations we might trace the interests and concerns of the industry to see how they shift and transform over the course of almost four decades. Yet this would be an extremely challenging history to write, in that the teapot's use is largely colloquial and often unexceptional. The teapot operates in the background as a neutral palette, a "good enough" approximation of a material thing, both immediately recognizable and invisible in its generic signaling of "object-ness." As such I am interested less in the teapot as an object on which research operates, than as emblematic of an entire practice of object making.

Object Worlds

Thus it seems crucial that we return once again to the teapot itself, prior to its instantiation as an standardized abstraction. What sort of an object is it, and how is this significant to its broader history? Perhaps most importantly it is an everyday object, a domestic object. It is the object that was ready to hand as Martin and Sandra Newell sat down to afternoon tea, and in that moment of recognition became transformed through a kind of presence into an object of study to be graphed and modeled mathematically. The significance of this nearness is precisely the way in which it exemplifies the kind of world building that computational simulation enacts. Only that which is knowable as

[97] Panofsky, Erwin. "Jan van Eyck's Arnolfini Portrait" in *The Burlington Magazine for Connoisseurs*. Vol. 64, No. 372 (Mar., 1934), pp. 117-119+122-127

present may become an object of study for simulation. What's more, the kinds of objects that become present are themselves loaded with cultural and historical significance. We might meditate, as Ann Sophie Lehmann does, on the significance of this particular teapot as an icon of mid-century German design, suggesting that it is not simple chance that brings Martin and Sandra Newell to select it as an object of study[98], but the teapot's context does not end there. What is the significance of Newell's status as a British researcher, transported to Utah as part of a wide-reaching effort to create an international community and center for excellence in a place as isolated as Salt Lake City, that he would even sit down to afternoon tea as the anecdote suggests. Perhaps most importantly, what is the significance of the teapot as a highly gendered and domestic object, found not in the lab but in the home and brought into a space in which almost no female researchers are historically remembered, to become an icon and emblem of an entire discipline?

The University of Utah was by no means an entirely male-dominated research center, and at minimum Elaine Cohen plays a significant role in the development of b-spline geometry in the early 1970s. Clearly Cohen was not alone in her significant role in the history of computer graphics. In photographs of Sutherland's class digitizing the now-famous VW bug model several female students are seen actively participating in the process [FIG. 34]. When one inspects the photographic collection of the Evans & Sutherland Computer Corporation it also appears that the entire staff responsible for the assembly of computer hardware at E&S in the early 1970s were women [FIGS. 35, 36]. Their presence here recalls the hidden labor of women at a time when women *were* "computers," in that they were responsible for entering punched card data and running the

[98] Lehmann, Ann Sophie. *Ibid*. 175.

machines that made computing work. Theirs was the labor of the material objects of computing, of the hardware itself. As Katherine Hayles writes of Janet Freed, assistant to the famous Macy Conferences on cybernetics,

> She was the one who presided over the physical transformations of signifiers as they went from taperecording to transcript to revised copy to galley to book. Others […] worried about content - but her focus was the materiality of the processes that make sounds into words, marks into books.[99]

Perhaps it is telling that the presupposed immateriality of computer graphics elides the same labor of circuits and soldering that these women make possible. Instead the women we find in this history are wives and partners; they offer teapots and, in more than one instance, their faces or bodies to be modeled and measured. This too is the material history of the teapot as digital object.

These complex political circuits are made particularly clear in the case of another standard object: a 512x512 pixel test image known as Lenna, which is widely used to test image processing algorithms such as those for compression or denoising. This image of a brunette woman in a floppy hat peering over her shoulder is in fact the centerfold of the November 1972 issue of Playboy magazine, cropped at the shoulders [FIG. 37].

> Alexander Sawchuck estimates that it was in June or July of 1973 when he, then an assistant professor of electrical engineering at the USC Signal and Image Processing Institute (SIPI), along with a graduate student and the SIPI lab manager, was hurriedly searching the lab for a good image to scan for a colleague's conference paper. They had tired of their stock of usual test images, dull stuff dating back to television standards work in the early 1960s. They wanted something glossy to ensure good output dynamic range, and they wanted a human face. Just then, somebody happened to walk in with a recent issue of Playboy. The engineers tore away the top third of the centerfold so they could wrap it around the drum of their Muirhead wirephoto scanner, which they had

[99] Hayles, Katherine. *How We Became Posthuman: Virtual Bodies in Cybernetics, Literature, and Informatics*. Chicago, IL: University of Chicago Press. (1999) p. 81.

> outfitted with analog-to-digital converters (one each for the red, green, and blue channels) and a Hewlett Packard 2100 minicomputer. The Muirhead had a fixed resolution of 100 lines per inch and the engineers wanted a 512 $\times$ 512 image, so they limited the scan to the top 5.12 inches of the picture, effectively cropping it at the subject's shoulders. […] Those three sets of 512 lines—one set for each color, created imperfectly on the spur of the moment, with no purpose in mind beyond the job at hand—would become a de facto industry standard.[100]

This story of happenstance is familiar, as an everyday object is transformed into the standard by which other images are tested and compared for decades; but as this last example illustrates, media standards bring with them a set of connotations – a politics – and often produce a kind of cultural history among technicians and testers that over time can become decoupled from their original context, or can come to produce a new context all their own.[101] As such these images, objects, and processes make clear what is at stake in the production of a given technology, and illuminate the materiality of systems in the cultural and technical context of their production - a materiality that often becomes highly encoded and effaced. Significantly these standards often combine and overlap - quite literally in some cases - as image standards like Lenna are used to skin object standards such as the Utah teapot. It is not that researchers imagine a scenario in which one might need to accurately simulate a 70s *Playboy* pinup on a mid-century teapot, but rather that the objects serve as functional stand-ins for all images and all objects at once.

---

[100] Hutchison, Jamie (May/June 2001). "Culture, Communication, and an Information Age Madonna". IEEE Professional Communication Society Newsletter (archive) 45 (3).

[101] By the late 90s several women in the field of image processing complained that the image, while not explicitly offensive, was indicative of a larger problem in the field in which women were excluded or alienated. The editor of the editor of the IEEE *Photonics* journal Sunny Bains went so far as to ban the image from the publication in 1997.

Conclusion

In 1989, the SIGGRAPH convention for research into computer graphics was held in Boston, MA, home to The Computer Museum, which has since been shuttered. On display were a number of artifacts from the museum's collection, including the original Melitta teapot that inspired Martin Newell to create his computer graphics icon, having been donated by Sandra Newell five years prior in 1984. Surrounded by iconic photos of the teapot from its fifteen year history, the exhibit functioned both as a deliberation on the place of the teapot in the field, as well as a vivid reflection of the field's most enduring concerns - geometry, light, texture, and surface. [FIG. 38] The teapot itself was housed in a small display box under which a number of switches were rigged. Next to the box was a CRT monitor of the same size displaying an image of the Utah teapot. Visitors were prompted to flip the switches under the physical teapot, which triggered a set of colored lights to shine on its surface. In response to the switch the CRT image would change to match the color and light of the physical model, offering its simulated pair. In what ways does the one approximate the other? Or perhaps, phrased differently, how is a teapot like a jug?

"To learn what nearness is, we examined the jug near by,"[102] wrote Hediegger, and indeed our proximity to a particular set of objects in the world both limits and defines that which is made knowable and therefore available for simulation. This is true not only for the category of objects themselves - be they teapots or *Playboy* magazines - but how we understand and in turn describe them. For Hedidegger "the jug's thingness resides in

[102] Heidegger, Martin. "The Thing" 169.

its being *qua* vessel."[103] We become aware of this holding nature only when we fill the jug, yet it is not the sides and bottom of the jug that does the holding. Rather the void they create, "[t]he empty space, this nothing of the jug, is what the jug is as the holding vessel."[104] And yet our teapot has no bottom and can hold no liquid, save perhaps for a liquid simulation. Stranger still, our ceramic teapot could just as easily be rendered *as* liquid: a translucent pot held impossibly together, reflecting light not off its surface but through it's bright clear body. The teapot can be any number of things the jug cannot, in that it may simulate most any form of object that may be made legible to the material structure of computer graphical simulation. If the jug's thingness resides in its being *qua* vessel, the teapot's thingness resides in its being *qua* object. It is precisely that objectness produced through technology that Heidegger derides, the thing reduced to objectness alone.

Yet while it is tempting to dismiss such simulations as mere representation and as lacking any material basis in the world, for Heidegger the thing is irreducible to what we represent of it. "[S]omething independent can become an object, when we represent it to ourselves […] Yet what is thingly in the thing [does not] consist in the fact that a thing becomes the object of a representation."[105] While the simulated teapot may not approximate the materiality of the original Melitta, it is anything but immaterial. At minimum it is an object supported by a vast material history of production, from the cultural context of its identification and digitization, to the code and algorithmic objects

[103] Heidegger, Martin. *Ibid*. 166.
[104] Heidegger, Martin. *Ibid*. 167.
[105] Heidegger, M. 1994. Einblick in das was ist, *Bremer und Freiburger Vorträge*, Frankfurt, Klostermann p. 5, as quoted in Harman, Graham. "Objects, Technology, and Things in Heidegger" Cambridge Journal of Economics 2009.

that engage it, and the hardware and circuits that produce it. Moreover it materially circulates across a broad network of influence, from its early distribution among researchers who would memorize its patch strings, to its circulation via early networking technology, and through popular media. Finally it is important to recall that computer graphics as a field is explicitly preoccupied with the question of the materiality of physical objects: what they look like, how they change over time, how they interact with other objects, how they change when they are bent and broken. While simulation is the process by which it reproduces this materiality, its orientation is nonetheless toward the material world.

It is certainly true that simulation is framed by the limitations of our subjectivity, but this orientation comes to transform our relationship with the materiality of technical objects on a much broader scale. After all, interactive simulation is not exclusively invested in representational interfaces, it is the means by which we have come to conceptualize and produce the physical world as well. As such a wide range of objects in our lived environment might well be understood first and foremost as computational, structured as they are by simulation. The sleek design of an iPhone is little more than a bevel and extrude operation in Autodesk Maya, and the curve of a car's hood has for over fifty years been informed by the shape of a Bezier curve tested through comprehensive simulation. While simulation may not allow us to access a material world independent from ourselves, it nonetheless engages that world, transforming it through a material process of things in relation. This transformation does not emerge *ex nihlio*. It is the product of decades of research into geometric modeling and visualization. The objects we see now as graphical simulations are shaped by the long and negotiated process though which computer science first developed a theory of the object itself. It was a shift away from procedural mechanization and toward a dynamic field of interactive objects, a transformation from a set of tools into a *medium* with a unique ontology, one that is oriented toward objects both real and virtual.