Visualization of Complex Automotive Data

Jeffrey A. Stevens
General Motors

Making complicated data easier to understand has always been a challenge. Four types of visualization applications (CAD, generalized, specialized, and custom) have successfully been used by automotive manufacturers such as General Motors to help meet this goal. Here are some ways that common processes can be developed for all types of visualization.

According to one definition, visualization is “a method of representing large amounts of complex data in ways that are easier to understand, analyze, and support decision making.”[1] Large automotive manufacturers such as General Motors (GM) use visualization with great success. Visualization requires combinations of manual and automated data harvesting. Users must find and access the correct digital data based on a correct configuration of the product to be visualized. In addition, each type of visualization, from real-time design reviews to marketing-based image rendering, requires data to be prepared with appropriate materials, backgrounds, and lighting. Each of these processes requires a different set of expert users and their own optimized process to create specific visualizations or views of data.

History of visualization at GM

For the past 25 years automotive manufacturers have increasingly used visualization to make it easier to understand complex data. Computer rendering of synthetic images and color coding of abstract attributes were two of the first types of visualization. At GM, the interest in visualization started with John Dill’s research on color-coded curvature[2] and Dave Warn’s research on lighting controls for rendered images that became AutoColor, a former GM internal rendering application.[3] Also, in the mid-‘80s, interest started developing in using computer rendering and animation more broadly with the introduction of the first system for automotive industrial design or styling called Alias Research AutoStudio. AutoStudio is still in use in GM as part of the Autodesk AliasStudio suite of tools.[4] The use of real-time visualization has also gained interest since the late-’80s with the introduction of the first commercial visualization workstation (Silicon Graphics’ SGI 4D70GT). At GM, real-time visualization started with a question from a clay model validation (milling) center about what a nonuniform rational B-spline (NURBS) surface looks like, and from research on real-time shading of trimmed B-spline surfaces.[5] This evolved into a surface quality visualization application called SQV and an early digital mock-up application called SurfSeg. Real-time visualization was expanded in the mid-90s with Randy Smith’s immersive virtual reality (VR) application for GM called VisualEyes.[6]

The evolution of visualization at GM has included the transition from internally developed applications to commercial off-the-shelf (COTS) systems that can be supported globally. These COTS applications include UGS Teamcenter Visualization (TcVis) for real-time design reviews and digital mockups (http://ugs.com), Autodesk Maya for photorealistic image rendering and animation, and RTT DeltaGen for real-time high-end styling presentations (http://rtt.ag). Styling presentations include near photorealistic real-time rendering of product data to help design managers understand the current state of the design and make decisions between design alternatives. GM Europe’s Opel brand further extended the scope of real-time visualization with the introduction of a vehicle configurator called Opel Car Creator at its automobile dealerships in 2005.[7]

Visualization categories

Visualization processes for each of the previously mentioned visualization types have evolved somewhat independently. It is my premise that each of these visualization processes has common elements and some of the data can be reused and repurposed instead of recreated. To understand the level of commonality within these processes it is helpful to classify the visualization types into four primary categories—CAD (source), generalized (broad), specialized (focused), and custom (unique), as Table 1 shows. COTS visualization applications are constantly evolving, which can alter the boundaries among the categories, especially as related to the examples in Table 1.

CAD (source) visualization

CAD (source) visualization includes the visualization capabilities built into today’s CAD systems such as UGS NX and Autodesk AliasStudio. CAD applications func-

 Published by the IEEE Computer Society 0272-1716/07/$25.00

tion as data authoring (creation and editing) systems and are the source of 3D data within the visualization process. They have historically had some level of image-rendering capability. Now, with the emergence of low-cost, high-performance graphics cards in PCs and workstations, real-time visualization has become an important part of the CAD process for parts and components designing. CAD visualization capabilities include visual feedback to guide design development. Figure 1 shows data quality checking related to the aesthetic and manufacturing fidelity of the surface data with color-coded curvature visualization and grid reflections to review highlights. It also shows clearance and interference checking of adjacent parts. Design-in-context, where adjacent parts and reference data are loaded from a data management system and visualized with the design data, is also part of the CAD process. Collaboration using CAD visualization is mostly peer-to-peer between two users of the same CAD system to focus on the completeness and quality of their designs. CAD visualization is much more integrated with the CAD creation and editing processes than any of the other types of visualization.

Table 1. Visualization categories and examples of each.

Category	Examples
CAD (source)	Design reviews Surface or data quality Clearance and interference Design-in-context (tied to product data management systems) Peer-to-peer (CAD to CAD) collaboration
Generalized (broad)	Virtual reviews (ad hoc and multi-CAD) Virtual assessments (integration) Virtual audits Virtual builds (assembly sequence) Virtual analysis (Clearance and interference, functional animation) Cross-functional team collaboration
Specialized (focused)	VR (immersive reviews) Styling presentations CAE analysis results Photorealistic rendering In-context animation (video background) Global management and team collaboration
Custom (unique)	Vehicle configurator Web experiences Brochure rendering Commercial animation Video games (branded content) Multiuser game collaboration

Generalized (broad) visualization

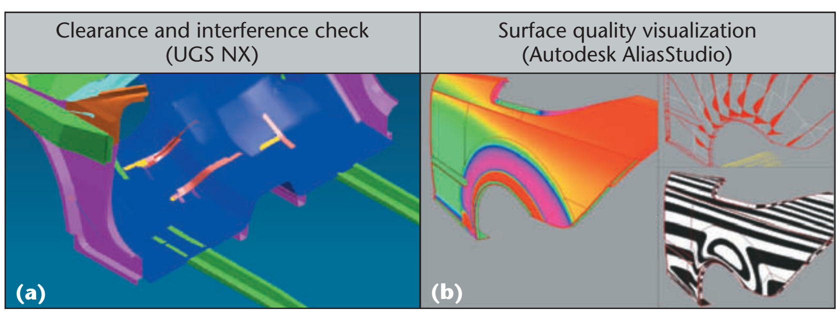
Generalized (or broad) visualization includes the capabilities of general visualization applications used across design, engineering, and manufacturing, and between original equipment manufacturers and their suppliers. It has the broadest scope of visualization usages and the largest amount of real-time visualization functionality. A large enterprise, such as an automobile company, might have more users of a generalized visualization application than of the core CAD application. TcVis is an example of a generalized visualization system.

Table 1 lists some examples of generalized visualization. A standardized visualization model containing both a polygonal representation and the surface representation from the CAD systems is the starting point for this type of visualization. The polygonal representation is typically checked or audited against relevant engineering and manufacturing criteria. Precise measurements are usually made with the surface representation. Parts in this visualization model can be checked against other parts, components, and assemblies to identify parts that are incorrectly designed or incorrectly positioned. Clearances and interferences among parts within vehicle compartments or among vehicle systems can also be assessed. Typically this analysis is performed with more data (larger data models) than a clearance check within the CAD systems. Engineers use assembly sequences to ensure that what appears to be a valid design can actually be built in a manufacturing plant.

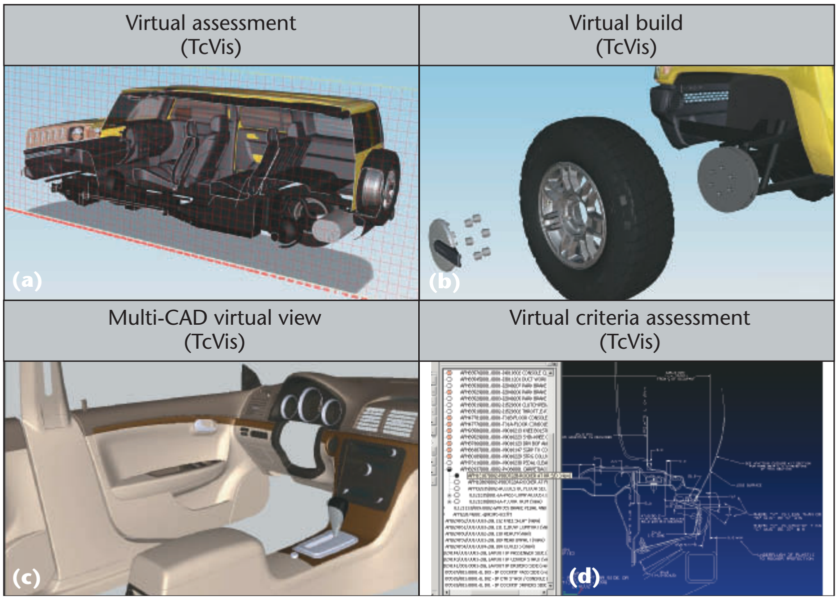
Data with moving parts can also be visualized and analyzed for interference or for ergonomic analysis. Cross-functional teams often collaborate using generalized visualization applications to conduct integrated reviews with team members. These reviews can be local within a site or they can be global between sites across the world. Generalized visualization applications are usually technology followers relative to visual image quality as they tend to use technologies that have been proven in specialized visualization applications. On the other hand, these same generalized systems are the technology leaders related to data integration, including the ability to access specific configurations from a data management system. Figure 2 (on the next page) shows a few examples of generalized visualization.

Specialized (focused) visualization

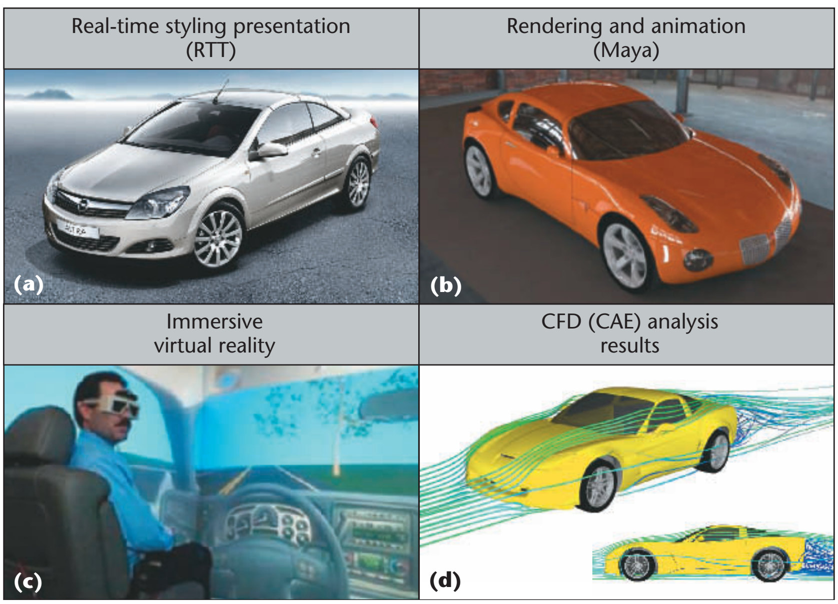
This category of visualization contains the largest number of distinct applications such as Autodesk Maya, mental images mental ray, RTT DeltaGen, Autodesk Showcase, CEI EnSight , and GM VisualEyes.



1 Examples of CAD (or source) visualization. (a) Clearance and interference check and (b) surface quality visualization.



2 Examples of generalized (broad) visualization, generated in Teamcenter Visualization (TcVis). (a) Virtual assessment, (b) virtual build, (c) multi-CAD virtual review, and (d) virtual criteria assessment.



3 Examples of specialized (focused) visualization. (a) Real-time styling presentation, (b) rendering and animation, (c) VR, and (d) computational fluid dynamics analysis results.

Many more specialized applications exist. Figure 3 shows some of the diversity in this category. Each system has a specific purpose and creates a focused view of the data. Specialized visualization applications are often technology leaders because of aggressive development within their focused areas. They have been the first to take advantage of new visualization capabilities such as GPU-supported programmable shaders; physically measured materials; image-based lighting; high-dynamic range images used as environments; and shadows based on occlusion culling, ray tracing, and global illumination. The visualization of computer-aided engineering (CAE) analysis and simulation results is another type of specialized visualization. This can include finite element analysis results for structural analysis or crashworthiness, computational fluid dynamics results, or performance simulation results. Over time, some of these specialized visualization capabilities become part of generalized visualization systems and even CAD applications.

Two other specialized visualization trends include transitioning from offline image rendering to workstation-based real-time visualization and the replacement of internally developed specialized applications with COTS software.

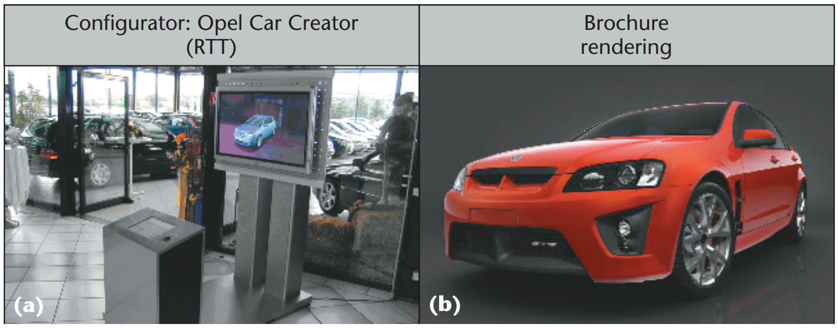
The different focuses of these specialized visualization systems demonstrate equally different user experiences: designed or crafted experiences, guided experiences, and shared experiences. With animation, a designed experience lets the end user experience what the author crafted. With real-time styling presentations the design management team is guided through views of data with the intent of making a decision. Collaborative immersive VR reviews can create a true shared experience.

Custom (unique) visualization

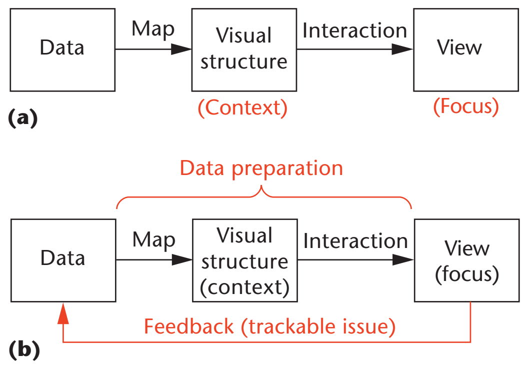
Table 1 lists some different types of custom (unique) visualizations. For example, a vehicle configurator lets customers select the vehicle and option content that interests them; they then can see their vehicle choice with real-time visualization. The rapid growth in computer games and the new graphics capabilities in game consoles have also created new visualization opportunities. In addition, multiuser games and experiential Web environments are new forms of collaborative visualization. Much of the custom visualization work is performed outside automobile companies using CAD data or visualization models provided by the companies. Figure 4 shows examples.

The visualization process

To better understand visualization processes it is useful to look at other areas of visualization such as scientific visualization or information visualization. Wakita and Matsumoto defined a "process of information visualization" where the originating data is mapped onto visual structures and an interaction is designed to create focused views of data.[8] According to IBM, "Visualization is the process of turning data into insight."[1] In automotive visualization, the mapping of data into visual structures and the designing of interactions for focused viewing is called the data preparation process. The visual structures can be visual formats, configurations (group, assembly, or hierarchy), or virtual contexts with defined environments. Feedback from the design reviews can also guide the data refinement processes within the source CAD systems. In addition, the issues discovered during these



4 Examples of custom (unique) visualization. (a) Opel Car Creator configurator and (b) a brochure rendering.



5 (a) The process of information visualization.[7] (b) The visualization of data flow.

reviews are often tracked until a resolution has been incorporated into the data. In Figure 5, I added the concepts of data preparation and feedback relative to the visualization data flow into this process model.

Analyses of visualization processes

Today, concepts such as process reengineering, lean design, and value stream management focus on eliminating waste. These concepts can be applied to some of the steps in the visualization process including data translation, data preparation, and data management. By analyzing six distinct types of visualization processes, we can identify the similarities and differences among the processes.

Surface quality visualization

Surface quality checking and the corresponding visualization of surface attributes have transitioned from standalone specialized visualization applications to features within standard CAD systems. Surface data such as NURBS, Bezier, and procedural surfaces can be tesselated to create a polygonal representation. Data translation might occur when the CAD system invokes the functions if the polygonal data is not already present within the data file. This representation or visual format can then be visualized with focused views such as real-time rendered images with reflected grid lines or color-coded curvature maps. The surface quality visualization process is integrated within the context of the part design or data creation process to achieve the desired level of quality.

Virtual assessments and builds

This type of visualization process is enabled with generalized visualization applications. Virtual assessments, builds, and reviews using UGS TcVis are based on an existing visualization model that represents the design, as Figure 2 shows. The visualization model can include both polygonal data and surface data. The visualization process starts with CAD data that is translated into the required visualization format. This is a push process where the data translation is triggered by a user request or a status assigned to the data. Generalized visualizations can integrate data from multiple CAD systems or data sources as long as they are translated to the same visualization format. Using configuration capabilities within a data management system, an assembly of parts can be pulled together into a visual structure. The creation of standard views and snapshots allows cross-functional teams from design, engineering, and manufacturing to explore and understand the state of the design.

Rendering and animation

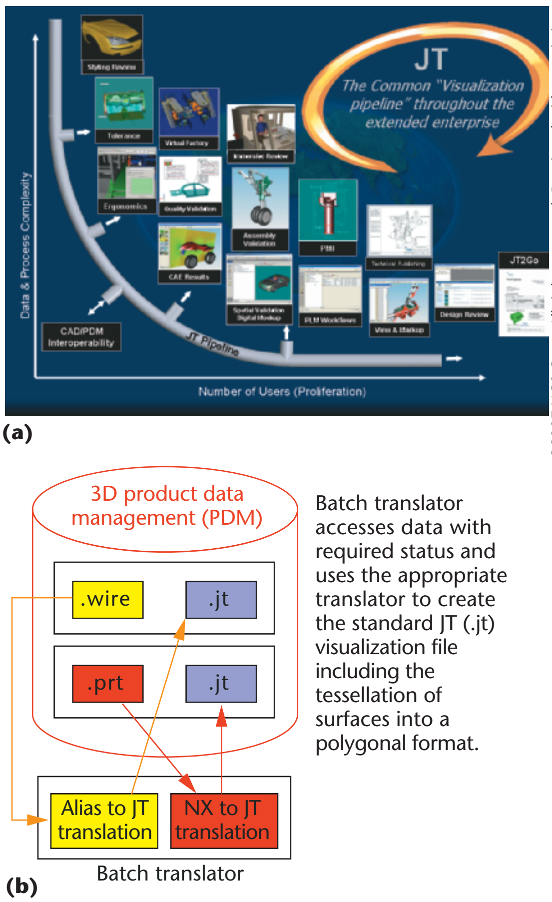
Photorealistic rendering and animation (see Figure 3b) is a pull process where the necessary data is pulled into the rendering or animation application piece by piece or part by part. The data preparation process includes scene development, material assignment, and lighting and shadows, specified for mental images mental ray rendering. Tessellation of the surface data could actually occur on a per-frame basis during the mental ray rendering process. The focused views are the final rendered images or the animated movie files. Systems such as Autodesk Maya can read source CAD files.

Styling presentations

The process for styling presentations uses tools such as RTT DeltaGen (see Figure 3a). Similar to rendering and animation, this is a pull process where the most optimized data flow involves reading the CAD data directly into the presentation system. The data preparation process is also similar in the creation of scenes, assignment of materials to parts, and the setting up of light sources to create the proper visual effects. Interactive behaviors can also be added to the data letting a door open or the wheels of the vehicle to turn. Predefined views or snapshots can also be added. The focused views are the real-time scenes and predefined views.

Immersive VR

The data translation for VR is a push process where the data is translated from the CAD systems by user requests and custom translators. The data preparation process is similar to the styling visualization data preparation with creation of scenes. The focused views are the real-time immersive scenes. VisualEyes, an internal GM VR application (shown in Figure 3), uses polygonal data converted from CAD systems.



6 **(a) Standard visualization format.[10] (b) JT translation.**

3D configurator

This is a custom type of visualization. Data is provided through a unique push process where the CAD system outputs a standard visualization file. The unique part of this process is that the data is still in the boundary representation (B-Rep) and NURBS surface format. The data is then tessellated as part of the data preparation. The focused views are real-time views of specific vehicles options. The Opel Car Creator is a point-of-sale dealership vehicle configurator with real-time visualization capabilities (see Figure 4).

Reengineering and a common visualization process

The six data processing workflows analyzed show different ways that a tessellated polygon model can be created. Both push and pull processes are used and the data can be tessellated at the beginning of the process (generalized visualization with a standardized visualization format), in the middle of the process (specialized visualization systems reading CAD files), or at the end of the process (animation with tessellation per image frame during the rendering process). The result is multiple visualization data flows with replicated versions of the visualization data. Similarities between these processes include the complexity of the data in terms of structure and hierarchy. Differences can include the size of the visualization model in terms of the number and density of polygons. In addition, the implementation of each type of visualization within the enterprise creates small distinct groups of visualization experts.

Several concepts and best practices exist that can be integrated to create a common visualization process that has the potential to eliminate duplicate data and duplicate efforts. These concepts include a standard visualization data format, levels of detail (LODs), data filtering, data configuration, virtual assets, virtual libraries of reusable content, and automating the data translation.

Standard visualization data format

An enterprise standard visualization data format lets all visualization capabilities start from the same data content. The UGS JT format[9] is an example of a standard visualization data format (as shown in Figure 6). Groups such as JT Open, a consortium supporting the development of the JT format, help to define the requirements for standard visualization formats. A standardized format is typically published and has appropriate toolkit support for accurately reading and writing the data files. In addition, a standard format must meet usability requirements. For example, the data must be able to be configured (assemblies of parts or surfaces with common attributes) and filtered to meet the needs of the required visualization. The contents of standard visualization files can include polygons, surfaces, attributes (that is, names, tolerances, and manufacturing information), and metadata (such as the release status). They can also include data organization concepts such as groups, layers, and parent-child hierarchies.

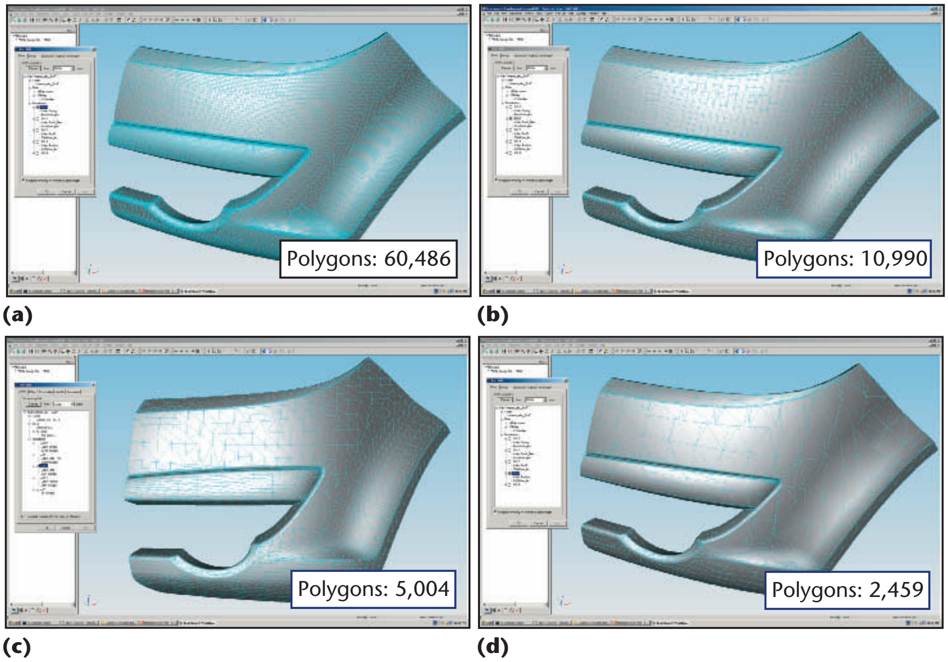
A concern with standard visualization formats is getting a broad enough adoption by the developers of the variety of visualization tools. Visualization applications need to be able to read and write the standard format.

Data translation

The creation of standard visualization data files typically involves data translation and the tessellation of CAD surface data into a polygon form. The translation's quality is directly related to the model's completeness, the tessellation's accuracy, and the resulting visualization's believability. The translated data needs to be complete with no missing parts and the tessellated polygonal model should have no visible holes or gaps. The distribution of the polygons in the mesh might be evenly distributed based on the underlying parameterization of the surface data. The distribution can also be refined based on the inherent shapes within the geometric model with more polygons in areas of high curvature and fewer polygons where the surfaces have less curvature or are flat. Some current product data management systems (PDMs) have built-in mechanisms for this type of data translation. When the user of a CAD system or the PDM system gives the managed CAD data a business-defined status (ready-to-be-shared, released, and so on), the PDM then tessellates the data and creates the standard visualization file, as Figure 6 shows. This tessellation can use the best available CAD tessellation algorithm or it can occur during the data translation using the capabilities of the standard visualization format toolkit.

Levels of detail

A LOD is a specific version of the polygonal model that is optimized for a specific purpose or usage of the data. The purpose can be a type of visualization such as a styling presentation or a factory fly through. The total amount of data to be visualized might impact the requirements for a LOD. The visualization's required image quality might also call for a specific LOD. Figure 7 shows the same set of surfaces with examples of four LODs. A chord height parameter can define how much a single polygon's interior can deviate from the mathematical surface definition. By varying the chord height value from 3.0 mm to 0.03 mm the number of polygons representing these surfaces can vary from approximately 2,500 polygons to 60,000 polygons. This means that complex data such as an automobile exterior or interior might contain several hundred thousand or even several million polygons. A single standard visualization data file can contain different levels of detail or multiple LODs. A visualization application might access LODs by reading in the most appropriate LOD based on the type of review or presentation. Access to LODs might also be based on a user's role or function. That user could be restricted from accessing LODs with too many or too few polygons.



7 Four types of levels of detail showing the same surface: (a) styling, (b) digital mock-up static, (c) digital mock-up motion, and (d) factory fly through.

LODs are also used within visualization applications to maintain an acceptable level of performance. When data is being moved or rotated the application might drop to an LOD with fewer polygons than when the data is static and there are no changes in the visualization. A styling presentation might only use one LOD to maintain the highest image quality even if sacrificing performance so that reflections and highlights can be assessed as the vehicle rotates. The use of different LODs might also depend on the data's distance from the visualization application's viewing window. Objects that are far away could have an LOD with fewer polygons and objects that are close could have the LOD with the best visual quality and the largest number of polygons.

In specialized and custom visualization applications that are more externally oriented (consumer and marketing) LODs tend to be handled differently than internal visualization applications. LODs are created for specific uses such as high-quality rendering for marketing brochures, animation for television commercials, Web specials, and 3D content for point-of-sale dealership configurators. Virtual models for each of these visualization applications tend to have only one LOD that has been optimized for that specific use.

A major concern with LODs is how many are needed for all of the expected uses of the data. Another question is how many LODs should be in one file and how many versions or renditions of the standard visualization file with a single LOD should exist.

Data filtering

Another concept in visualizing complex data is filtering out unnecessary content. The standard visualization format might include many types of data including B-Rep geometry, multiple LODs, manufacturing information, and general attributes. The visualization applications might only use some of this information. The filtering can occur as the file is being created if it is only used for a specific visualization. Utility programs can strip out content that should not be in the version of the files to be used. In addition, the filtering can occur when the standard visualization file is read into a specific application. The application might only read the information it needs or can use.

Data configuration

A common issue for most visualization applications is having the correct data to visualize. This can be data that has been released and is ready for downstream use. Styling data might be ready for criteria evaluation or surface quality audits. Engineering data might be ready for production. The correct data can also be the most current work-in-process data showing the latest changes to the design. For complex products such as automobiles it is critical to have the data configured to a specific model and variant or trim level. This could mean extracting all parts related to a coupe or convertible from a data structure that has all of the parts for all vehicles using the same platform or architecture. The data configuration must also be repeatable. It is critical to reassemble exactly the same model to correct an issue that was identified in a design review. It is also important to assemble a new model with the same recipe while replacing old files with newer ones to analyze whether the design changes corrected the problem.

A concern with current data configuration capabilities is that they work well only for CAD visualization and generalized visualization. If the product data is managed within a PDM system the configured model will need to be exported out of the PDM for use in specialized and custom visualizations.

Reusable virtual assets and libraries

Another concept is utilizing virtual assets within the different types of visualization applications. A virtual asset is a configured visualization model with a date or time stamp representing when the data was assembled and prepared. Once the virtual asset exists, the model does not need to be recreated but it can be reused. The data preparation process is simplified because the starting data model is the same for each process. Additional data preparation work is still required to meet the needs of the unique focused viewing but much of the duplicate work has been eliminated or reduced. Virtual libraries allow content to be used by a broader set of users and shared across a variety of visualization applications. Libraries can contain specific material definitions, standard lighting, image environments, geometric environments, and any other type of data that can be used to add value in a visualization or visual presentation.

One of the primary concerns about reusable virtual assets is how they will be managed. Assets tend to be complete representations. A newer asset might replace an older asset for a new visualization. The data management requirements might not be the same for complete virtual assets as they are for configurable product data.

One of the concerns about virtual libraries is that the description of materials is not common across visualization applications. A material might be defined by the parameters needed to render it using the standard real-time visualization capabilities of a PC. It might also be defined by a custom language or graphical description (programmable shader) that can be rendered with the newest graphics cards in a high-end workstation. A third version of that material might be defined only with parameters used by a software rendering algorithm. It might also be directly measured from a physical object or fabric.

A potential area of research is to define shareable materials that can be used within a variety of applications and have the same or similar visual effects. It might be necessary to have separate renditions or versions of a named material so that each visualization application can use the most appropriate definition for that material.

Conclusion

This paper introduced four categories of visualization applications to better understand the different types of visualization data flows. Concepts were introduced to help commonize steps within these visualization processes. Future directions related to the four categories include looking at the potential integration of the visualization data flow processes. This will involve analyzing the data models used for each of the visualization categories. Relationships between these data models need to be identified and the data management requirements for each of the models will also need to be defined. ■

Acknowledgments

I wish to thank David Evans, Shirish More, Alan Rhodes, Thomas Sahlin, Jim Johnson, Radhika Cherukuru, Thomas Heermann (Autodesk), Mike Zink (UGS), Martin Wood, Jens Rudnick, Ralf Hess, Tim Reynolds, Tom Celusnak (EDS), Ken Versprille (CPDA), and Dan Jankowski for their help in providing images and reviewing this article.

References

1. "IBM Deep Computing Visualization," white paper, IBM, Jan.2005; http://whitepapers.techrepublic.com.com/whitepaper.aspx?docid=148968.
2. J. Dill, "An Application of Color Graphics to the Display of Surface Curvature," *Proc. Siggraph,* ACM Press, 1981, pp. 153-161.
3. D. Warn, "Lighting Controls for Synthetic Images," *Proc. Siggraph,* ACM Press, 1983, pp. 13-21.
4. A. Rhodes and J.A. Stevens, "The Global Workflow from Design to Manufacturing: A GM Perspective," Pace Annual Forum, 2006; http://www.pacepartners.org.
5. W.H. Frey et al., "Color Shaded Displays of Trimmed B-Spline Surfaces," GM internal document CS 586, 1989.
6. R.C. Smith et al., "Really Getting into your Work: The Use of Immersive Simulations," *Proc. First Symposium on Virtual Reality in Manufacturing, Research, and Education*, 1996, pp. 1-10.
7. R. Hess, "Opel Car Creator: 'Selling without Limits'," *Proc. 7th Real Time Technologies Conf.*, RTT, 2006.
8. A. Wakita and F. Matsumoto, "Information Visualization with Web 3D," *Computer Graphics*, vol. 37, no. 3, 2003, pp. 29-33.
9. JT File Format Reference First Edition V8.12006; http://www.jtopen.com/technology/format_reference.shtml.
10. M. Matousek, "Innovation is a Team Sport." UGS, 2005; http://ugs.com.

Jeffrey A. Stevens *is a global styling and visualization technologist at General Motors. His research interests include CAD, curve and surface development, visualization, VR, automotive industrial design, and technology strategy. Stevens has an MS in engineering science from Rensselaer Polytechnic Institute. Contact him at jeff.a.stevens@gm.com.*