SDC-90-00064

SDC
SOLENOIDAL DETECTOR NOTES

CAD/CAE SYSTEM EVALUATION

D. Shuman

July 3, 1990

Lawrence Berkeley Laboratory - University of California **Engineering Note**		Code SD-8103	Serial M7027	Page 1 of 9
		WBS Code 8.1.3-S	SSC Det # 90-00064	CD #
Author D. Shuman	Department Mechanical Engineering		Date 7/3/90	
Program **Solenoidal Detector Collaboration**				
Subprogram **Subsystem Integration**			**LIMITED ~~DISTRIBUTION~~**	
Title **CAD/CAE System Evaluation**				

CAD/CAE requirements and software evaluations

General

It is difficult to determine which CAD/CAE system is best for a particular project, especially one in an initial development stage. Detector design integration tasks may well need more than one type of system, as architectural, civil and mechanical tasks are present. It is our feeling that a good mechanical engineering oriented solid modeling system will be most useful for initial work. Thus we limited ourselves to reviewing only the top-rated solid-model based drafting and engineering software packages which run on different hardware platforms. There were four software packages which fit the bill for evaluation: Parametric Technologies Pro/Engineer, Unigraphics UGII with Uni-Solids, SDRC's I-DEAS, and HP's ME30, which is already in use at LBL. Both SDRC and Unigraphics are being used for SSC work; Parametric Technologies is relatively new and, and as far as we know, hasn't been evaluated yet by the SSC community. The main evaluators of the software packages were myself and Gabby Obegi, LBL Mech. Engineering CAD manager. Valuable input was also obtained from Bill Edwards and Yoshy Minamihara.

Evaluation Summary

We feel that Parametric Technologies (PT) Pro/Engineer software has the strongest capability to perform the subsystem integration CAE tasks. The parametric capabilities extend across the various software routines, are an integral part of the database, and work especially well in modeling of assemblies. Its rigorous scheme for defining geometry and assemblies should substantially reduce errors. It has limited but sufficient IGES capabilities with IGES 4.0. It is the easiest to use, both engineers and some designers will be able to use it, even part time. It has excellent finite element meshing capability and will have, in the future, a bi-directional link to our current finite element software, ANSYS. Its drawbacks are that it is new software, without the wide range of features found with the other software packages. It lacks a macro language and advanced graphic display features. Updates have been substantial and frequent however.

Both UniGraphics' UGII and SDRC's I-DEAS software suffer from having cumbersome solid modelers, incomplete 3D-2D associativity, non-parametric assembly definition, confusing and "theoretical" user-interfaces. In addition SDRC has limited IGES capability.though this should change soon with support for the IGES 4.0 standard. They are both good software packages overall, and represent many years of development and refinement, and have been extensively de-bugged. There are large support staffs available and the companies are very stable. Updates do not come very often, on a yearly basis at best. ME30 is a very easy to use system but lacked support for IGES 3-D input which removed it from further consideration.

In short, PT represents the state of the art in CAE concepts, which is strongly evidenced by both UG and SDRC (as well as most other CAE vendors) rushing to develop their own version of parametric CAE. Both UG and SDRC represent older technology that has been extensively developed, featured and de-bugged. PT represents a certain risk, in that it is a new product, and its capabilities and shortcomings are not well known. In our short period of evaluation based on vendor demonstrations, it is impossible to know if the bugs and limitations which we were not able to find will outweigh the positive features we have found. We feel this is a good time to buy it and evaluate it; should it not turn out to be the right product, we will have gained much insight into the particular needs of the integration effort regardless. LBL Mechanical Engineering is very interested in the product for their own projects, and would, in all probability, buy the license from SDC should we wish to switch software after this evaluation period.

Finally, it became clear to us that, whichever system we chose, much work remains to be done to set-up and customize it to fit our needs. Working with the vendors will remain an on-going task. IGES translations, in particular, will require a large effort to facilitate the many different CAD systems in the collaboration and ensure they are all talking the same IGES "language".

Integration CAE Tasks (as foreseen)

Control Volume Definition and Interference Control; Interface Design

The SDC Integration Steering Committee and Technical Board (ISC/TB) will help design control volume geometries for the various subsystems, and will assure that subsystem designs fit within them. The ISC/TB will also keep track of the mechanical, fluid and electrical interfaces between subsystems. The ISC/TB will provide IGES or proprietary format volume geometry files (along with plotted drawings). It is hoped that collaborators will, in turn, provide IGES or proprietary format surface/solid models which can be checked against ISC/TB control volumes. IGES 4.0 compatibility may be required in several months (see section on IGES below). Modifications to geometries will be negotiated as part of the development process. ISC/TB will provide files showing interferences between subsystems.

Assembly sequence modeling.

The assembly of major subsystem components will be conceptually modeled, with movement sequences analyzed for interferences and other non-feasibilities. Ideas for procedures and associated lift fixtures will be developed from these studies. Picture sequences will be created to clearly show the various assembly and maintenance schemes.

Support Structure Design and Analysis

The SDC integration effort at LBL (SDCI) may explore schemes of supporting major subsystem parts, and may be required to develop, analyze, and optimize needed support structures. Finite element analysis may be used for complex structures. These structures will be integrated into an overall layout and SDCI will coordinate the incorporation of needed features of each subsystem. SDCI will keep track of the structural interfaces between major components.

Support Facility Layout

SDCI will develop layouts for both surface and subsurface facilities such as assembly areas, shops and clean rooms, counting house(s), computing rooms, power supplies and switching houses, refrigerators, dewars and bottle racks, etc. These will be integrated with SSC experimental area layouts in conjunction with SSCL. The building design will be the responsibility of the SSCL AE subcontractor.

Piping and Electrical Routing

SDCI will manage the routing of all piping and cabling in and out of the various subsystems. Interfaces will be defined. Parts lists will be generated showing pipes, flanges, high current busses, signal cables, etc.

CAE System Performance

IGES Compatibility and Capability

It must be said from the outset that the idea of using (3D) IGES (Initial Graphics Exchange Format) may, as it has in the past, prove to be unworkable as an efficient means of communication between software packages. 3-D solid model software packages are sufficiently different in the way model data is created and organized, that the IGES format is simply insufficient to communicate the full "intelligence" of the model. Many software vendors simply write their own translators which bypass IGES altogether, and write directly to another (single) software package.

At a miminum we seek to both receive and output unambiguous surface models, so we specify that the software must be able to input and output IGES 4.0 files(which support trimmed surface definition). The software should be able to build a true solid from these 4.0 files, largely on its own. Since both UG and PT are boundary representation (B-rep) solid modelers, they are able, in theory to create a solid model from a closed surface model (which we have seen demonstrated on both systems). Since SDRC uses constructive solid geometry (CSG), rather than B-rep to build true 3-D models, it cannot, on its own, recreate a solid model from a given surface model, which may be a serious drawback (one must manually recreate the solid, using the surface model as an input aid).

For a similar reason, ME30 doesn't have the ability to import IGES 3-D files at all, which removes it from further consideration.

A possible future requirement for SDC collaborators may be the capability to output and input IGES 4.0 files as IGES 3.0 does not support "trimmed" surfaces. Without this capability, files usually contain extraneous surface information that is indistinguishable from real surface data. This will seriously complicate integration tasks such as interference checking and subsystem interface definition, possibly leading to numerous errors. Note: SDRC Ideas does not yet support IGES 4.0 on HP workstations, and a recent SDRC IDEAS 4.1 IGES 3.0 file from Fermilab translated into Unigraphics UGII with extraneous surface data that appeared to be indistinguishable from real surface data .The model could not be turned into a solid, however (the IGES 3.0 standard will not even allow SDRC's IDEAS 4.1 to correctly read in its own IGES output!). The next release of IDEAS (5.0) should support IGES 4.0. Since UG is both a surface and a solid modeler it was able to display the resulting strange surface model. PT cannot read in IGES 3.0 files that contain untrimmed surface data (the same FNAL file), as it tries to create a solid immediately, which is impossible with untrimmed surfaces. They will be able to accept wireframe-only data with their next release. They can output and input IGES 4.0 files from themselves fairly well, creating a solid from the (trimmed) surface data. The system flags ambiguous surfaces and prompts the user to define the surface normal direction, which seems to work well. We don't know whether it will close small gaps automatically, as does UG When solids are created from IGES with PT, the resulting solid is not modifiable or parameterizable. One can, however, add features and subsequent parameters to the model.

Assembly Modeling

PT has intuitive assembly capabilities and the ability to do substitutions parametrically. This will be useful for substituting detector subassemblies with their control volumes and doing interference checking. It is the only system seen (we haven't yet seen SDRC's assembly capabilities) which preserves assembly relations after part modification. The user establishes assembly relations that mate, align, insert, etc. one part's features to another part's features (one can also use fixed coordinates). This is a powerful feature for assembly, as it prevents errors of interference, if the user is smart enough to establish the relationships needed. Automatic exploded views are easy to create and explode distances are easily modified.

UG's Assembly capability is of limited usefulness. One can edit parts (actually a copy of a part) in assembly environment, however, if any 2-D drawings have been created with the original part, they are simply discarded when the revised model is updated. Functionally, this is a lack of associativity in the assembly environment. Part placement in assembly is fair, as feature coordinate systems can be used as handles to match the part up with other part feature coordinate systems. This is difficult to visualize for many people, and unnecessarily "theoretical". Worst of all, if parts are modified, the orientation scheme is lost (no design intent is preserved) because assembly is not parametric, or feature-to-feature based. One must reposition parts and perform a visual or interference check. UG does not have an automatic exploded view capability.. UG has no control volume substitution or assembly management tools as does PT and SDRC. UG has no part instancing; when a part is used in an assembly the geometry is copied into the assembly drawing, thus losing associativity with the actual part file. PT assembly files always reference actual part files, maintaining associativity, and reducing file size.

Interference Checking

PT does not show volumes of intersections, only subtractions or additions. In theory this is OK, but subtractions need to be clearly delineated on complex structures, which is not the case. Meanwhile, interference checking is difficult to do, due to poor visualization of interfering surfaces. A macro for automatic checking may be needed, even though such macros have long run times. It is unclear whether PT's programming capabilities will allow the writing of such a macro. PT calculates interference dimensions and prints them out. UG has no interference dimension printout, only affected face display. UG easily shows volumes of intersection, and works fairly straightforward. Using UG's shaded working image, users can easily pick objects to be checked and work in shades or hidden line mode, and the model can easily be rotated to allow easy pointing. UG has a macro for automatic checking. SDRC has a number of cross-assembly checking routines which may simplify the process greatly. These are routines which systematically check each part in an assembly against each other, check subassemblies against each other, etc.

Parametric Geometry and Assemblies

This is where PT has a decided advantage. PT has made a parametric database the core of their software; almost everything is parameterized, including assembly relations. Every dimension in a PT model is a variable, and can be a function of other previously set dimensions or parameters, e.g. pressures, loads, etc. A very simple example would be to define a calorimeter depth as a function of beam energy. Or perhaps the I.D. of a barrel calorimeter is defined as the O.D. of a coil cryostat plus a set clearance (or vice-versa). Once defined, if the coil O.D. is changed, the calorimeter I.D. changes automatically, thus avoiding an interference. In PT, there is no "looping" allowed (e.g. $a=f(b)$, where $b=f(c)$, where $c=f(a)$) which maintains a definite hierarchy to the dimensioning scheme. This is good for tasks such as ours, where we want to maintain an unambiguous dimensioning scheme, however, it is not so useful for loosely defined schemes where the dimensions need to "float for a while" or optimization needs to be done (e.g. linkage synthesis). Dimension schemes are re-orderable and modifiable; the system notifies the user when conflicts or ambiguities arise. Many obvious questions arise when pondering the nature of this parametric model definition, which will need to be worked out.

Only SDRC has something similar which they call "variational geometry". This feature is limited to 2D geometry. Though the mathematical foundation for variational geometry is sound, it appears difficult to extend the concept to 3D without making extreme demands for computing power. The system works best for optimizing linkage geometry and solving engineering problems, such as iterative geometry solutions.

UG is developing a software module said to combine the best features of the various parametric systems, but this is not due out until '91. A key question is whether UG will be able to integrate parametrics fully with their current (non-parametric) data structure.

Ease of Use and Program Logic

It should be said at the outset, that no solid modelers are easy to use, period. PT has the most logical and unified environment and is clearly easier to use than the other two systems. The downside of this is that the environment may be unnecessarily restrictive for unusual tasks, such as IGES translations, or hermeticity analyses. Prompts are present on almost every command, though they are rather short. 3D geometry creation, although involved, is very consistent and straightforward, especially with the sketcher. The sketcher allows one to sketch geometry profiles in a crude manner, then dimension them later. This is much faster and easier to visualize than drawing everything exactly as dimensioned immediately. In addition the sketcher can make a number of assumptions about the design intent, saving time. With PT, one is always creating a solid, one does not create wireframes or surfaces, then turn them into solids. Commands follow a logical thought process; there is no need to set certain parameters up beforehand, which is a major problem with some software. There is a built-in help window available. In general, PT seemed the easiest to use and some designers will be able to use it along with engineers. On the other hand, some engineers may also be frustrated, as they are "locked-in" to a methodology and can't just "do what they want to do".

UG requires coordinate point type-ins as geometry is created, which requires much mouse to keyboard switching and slows geometry creation. UG is working on their own version of a sketcher. Prompts were not present; one must be very familiar with the system to know what it wants next. There is no help window. This is an engineers-only workstation, with a wide range of features to be learned. The macro capability should allow engineers to solve most difficult problems.

SDRC seemed hardest to use, as it was hard to tell where in the menu one was at most times. Cascading submenus without prompts force the user to read all the menu picks to figure out which one is appropriate. Cascading menus (from my past experience with Intergraph) are quickly tiresome due to the mouse gymnastics which must be mastered to move quickly through them. The flashing of menu windows on and off is distracting. There was a consensus that this system would be difficult to use. Again, this is an engineers only workstation. There is a help window, however.

Drafting and 2D Associativity

Only PT has full 3D->2D->3D associativity, where the associated drawings are changed automatically after editing the 3-D model, or the model may be changed by changing the drawings. In PT, parts may be modified in any environment: design, drafting, or assembly. Assembly relationships are preserved after modification. SDRC drafting is poor, with a top-down" approach to drafting (no model editing, sectioning, hidden line rotations, view setups in the drafting environment). SDRC requires draftspeople to go to solids environment in order to do initial view layout, then go to view layout environment to pass views into drafting (confusing and restrictive). SDRC has a rudimentary form of 3D->2D associativity (changed dimensions are flagged in red). Parts may not be modified in drafting environment (no 2D->3D associativity). UG has a serious deficiency whereby part changes made in assembly mode (other parts visible) are not file-able unless one wants to discard all associated 2D drawings of the part. Part modification in assembly mode is very useful, as one is often designing parts to either fit or stay clear of other parts. In essence, 3D->2D->3D associativity does not exist across all of UG's environments, it only exists in a limited area.

Graphics Display

UG seems to have the most developed versatility in this area. UG has the capability to actually work with the shaded image or a hidden line image while model editing. One can also display a superimposed hidden line, shaded and wireframe image. UG does not have more than one light source on the HP-SRX graphics processor; neither does PT. PT can only be used in a hidden line or wireframe mode; shading is a display only function. PT's graphics in general need further development, however we should be able to produce quality shaded images with different colors. SDRC utilized the 7 light sources available on the SRX box, but we did not see them. So far, we have seen only UG used on HP hardware, with the SRX graphics processor, though all three vendors claim to support it. PT does not sell many seats on HP hardware, but we verified a reference who claims it does work well on HP hardware. We did not see a good demonstration of the various SDRC graphics capabilities. All three systems have, at extra cost, software shading routines to produce high quality shaded images, at the expense of computing time. Other outside software also exists with full ray-tracing routines, radiosity functions, etc.

FE compatibility

FNAL, ANL, LBL and many other engineers use or are familiar with ANSYS, so systems with strong links are preferable. PT has signed agreements with ANSYS to develop a bi-directional link their two products within the next year which promises very powerful capabilities. An interface already exists which allows passing meshed (hard) geometry into and out of ANSYS, similar to UG, however, the new interface will allow the parameterized models to retain their parameters in ANSYS, creating tools for automatic design optimization. SDRC does this already but not with ANSYS, and they have no plans to develop such a link. UG will have to develop their UG/Concept before they can develop these links. For meshing in general, we saw the best demonstration with PT's mesher. Local mesh refinements were very easy to do. SDRC supposedly has powerful mesh refinement techniques, but we did not see them. UG has limited mesh refinement capability. UG interfaces to PATRAN (a pre- and post-processing program) well, which then integrates to other FE programs such as NASTRAN, ANSYS. This might be a good system, as PATRAN is reputed to be a very good pre- and post- processor, however one must buy PATRAN separately (cost is high, but unknown)

Macro Language Capability

Situations often arise where basic system functions are incapable of performing the desired task, or the steps necessary are long and tedious. A macro programming language is necessary to automat these tasks. UG has a language called GRIP which seems very capable and complete. UG also supports a computer bulletin board library containing over a thousand GRIP programs written by both UG and their customers. Though most of time these programs don't do what you want them to, they can often be copied and modified in much less time than it takes to write one from scratch. PT has no internal macro language as of yet, however it does have the capability to execute user programs written in C. This is both an advantage and a disadvantage; though C is a fast, powerful language, the accessibility of geometry creation and system functions is a big unknown. Programs must be compiled before running and debugging may be very hard to do. A macro language would be preferable. We did not inquire about SDRC's macro capability, due to time limitations.

Price

Each vendor sells the software in modules, to allow tailoring systems to fit customers needs and budgets. The following prices are based on software configurations which include the following basic capabilities:

1. 3D solid modeling
2. Assembly of parts
3. Finite element meshing of shell structures minimum
4. Parametric design capability
5. Plotting to HPGL plotters
6. IGES 4.0 input and output

7. 2D detail drafting
8. Network operating license with X-window capability

Prices were:

1. PTC's Pro/Engineer	$19,500
2. UniGraphics UGII	$34,000
3. SDRC's I-DEAS	$26,000

In general, we felt that even though there is a wide price disparity, the prices a still low enough that price should have as little effect as possible on our choice. We felt that needed capabilities are worth paying for. Since Pro/Engineer is a new product and is trying to compete against the "big guys" the above price is probably lower than it will be in the future, when and if PT joins their league, as we believe it will. Maintenance cost on all three systems ranges from $4k-$5.5k per year