

Final Report (Year #1 only)
for

CAPRI:
Using A Geometric Foundation for Computational Analysis and Design

Robert Haimes
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
haimes@orville.mit.edu

Introduction

The computational steps traditionally taken for most engineering analysis (CFD, structural analysis, and etc.) are:

- Surface Generation -- usually by employing a CAD system
- Grid Generation -- preparing the volume for the simulation
- Flow Solver -- producing the results at the specified operational point
- Post-processing Visualization -- interactively attempting to understand the results

For structural analysis, integrated systems can be obtained from a number of commercial vendors. These components couple directly to a number of CAD systems and are executed from within the CAD GUI. It should be noted that the structures problem is more tractable than CFD; there are fewer mesh topologies used and the grids are not as fine (this problem space does not have the length scaling issues of fluids).

For CFD, these steps have worked well in the past for simple steady-state simulations at the expense of much user interaction. The data was transmitted between phases via files. In most cases, the output from a CAD system could go IGES files. The output from Grid Generators and Solvers do not really have standards though there are a couple of file formats that can be used for a subset of the gridding (i.e. PLOT3D data formats and the upcoming CGNS). The user would have to patch up the data or translate from one format to another to move to the next step. Sometimes this could take days. Specifically the problems with this procedure are:

- File based. Information flows from one step to the next via data files with formats specified for that procedure. File standards, when they exist, are wholly inadequate. For example, geometry from CAD systems (transmitted via IGES files) is defined as disjoint surfaces and curves (as well as masses of other information of no interest for the Grid Generator). This is particularly onerous for modern CAD systems based on solid modeling. The part was a proper solid and in the translation to IGES has lost this important characteristic. STEP is another standard for CAD data that exists and supports the concept of a solid. The problem with STEP is that a solid modeling geometry kernel is required to do anything with this type of file.

- 'Good' Geometry. A bottleneck in getting results from a solver is the construction of proper geometry to be fed to the grid generator. With 'good' geometry a grid can be constructed in tens of minutes (even with a complex configuration) using unstructured techniques. Adroit multi-block methods are not far behind. This means that a multi-million node steady-state solution can be computed on the order of hours (using current high performance computers) starting from this 'good' geometry. Unfortunately, the geometry usually transmitted from the CAD system is not 'good' in the grid generator sense. The grid generator needs smooth closed solid geometry. It can take a week (or more) of interaction with the CAD output (sometimes by hand) before the process can begin.
- One-Way Communication. All information travels on from one phase to the next. This makes procedures like node adaptation difficult when attempting to add or move nodes that sit on bounding surfaces (when the actual surface data has been lost after the grid generation phase).

Until this process can be automated, more complex problems such as multi-disciplinary analysis or using the above procedure for design becomes prohibitive. There is also no way to easily deal with this system in a modular manner. One can only replace the grid generator, for example, if the software reads and writes the same files.

Instead of the serial approach to analysis as described above, **CAPRI** takes a geometry centric approach. This makes the actual geometry (not a discretized version) accessible to all phases of the analysis. The connection to the geometry is made through an Application Programming Interface (API) and NOT a file system. This API isolates the top-level applications (grid generators, solvers and visualization components) from the geometry engine. Also this allows the replacement of one geometry kernel with another, without effecting the top-level applications. For example, if UniGraphics is used as the CAD package then Parasolid (UG's own geometry engine) can be used for all geometric queries so that no solid geometry information is lost in a translation. This is much better than STEP because when the data is queried, the same software is executed as used in the CAD system. Therefore, one analyzes the exact part that is in the CAD system.

CAPRI uses the same idea as the commercial structural analysis codes but does not specify control. Software components of the CAD system are used, but the control of the software session is specified by the analysis suite, not the CAD operator. This also means that the license issues (may be) minimized and individuals need not have to know how to operate a CAD system in order to run the suite.

The CAPRI API

CAPRI is a software building tool-kit that refers to two ideas; (1) A simplified, object-oriented, hierarchical view of a solid part integrating both geometry and topology definitions, and (2) programming access to this part or assembly and any attached data.

A complete definition of the geometry and application programming interface can be found in the document "**CAPRI: Computational Analysis PRogramming Interface**". In summary the interface is sub-divided into the following functional components:

1. Utility routines -- These routines include the initialization of **CAPRI**, loading CAD parts and querying the operational status as well as closing the system down.
2. Geometry data-base queries -- This group of functions allow all top level applications to figure out and get detailed information on any geometric component in the Volume definition.
3. Point queries -- These calls allow grid generators, or solvers doing node adaptation, to snap points directly onto geometric entities.
4. Calculated or geometrically derived queries -- These entry points calculate data from the geometry to aid in grid generation.
5. Boundary data routines -- This part of **CAPRI** allows general data to be attached to Boundaries so that the boundary conditions can be specified and stored within **CAPRI**'s data-base.
6. Tag based routines -- This part of the API allows the specification of properties associated with either the Volume (material properties) or Boundary (surface properties) entities.
7. Geometry based interpolation routines -- This part of the API facilitates Multi-disciplinary coupling and allows zooming through Boundary Attachments.
8. Geometric creation and manipulation -- These calls facilitate constructing simple solid entities and perform the Boolean solid operations. Geometry constructed in this manner has the advantage that if the data is kept consistent with the CAD package, therefore a new design can be incorporated directly and is manufacturable.
9. Master-Model queries and part regeneration -- The part of the API (under construction) allows for the integration and modification of parameters associated with the CAD Master-Model representation of a Part or Assembly. See the attached document "Master-Model Interface Specification" for details.

CAPRI's Standing

CAPRI is becoming mature. The reader components (all of the API except the geometric creation and manipulation section) will be released as Rev 1.0 by the end of the calendar year. The testing of **CAPRI** is complex because of the large number of possible CAD systems and workstation combinations. The current **CAPRI** CAD/Workstation test matrix is:

	Parasolid	ProE	I-DEAS	CATIA V4	CV	Native - Felisa
Alpha	X					X
HP	X					X
IBM RS6K	X			X		X
SGI	X	X	X	X	X	X
SUN	X					X
LINUX	X					X
Windows NT/2000	X	X	X		X	X

CATIA V5 is under examination, but it is not clear what is the best approach for the programming interface. An AutoCAD geometry reader has not yet been implemented.

A CV (CompterVision's CADD S V) interface has been written in support of NPSS work with Allison/Rolls Royce and ICEM-CFD.

The most significant recent change for **CAPRI** is the ability to perform geometry creation and modification. This has prompted the addition of Boolean operations on solids to the API. These functions allow for the specification of fluid passages where the blade is the solid. The blade is simply subtracted from the passage to get the geometry for the CFD calculation. In general very complex shapes can be obtained through a few operations. The current status is:

	Parasolid	ProE	I-DEAS	CATIA V4	CV
Simple Solid Creation	X		X	X	X
Subtraction	X	X	X	X	X
Intersection	X	X	X	X	X
Union	X	X	X	X	X

The CV port currently has a problem writing the resultant part(s) out. There is also a problem with CATIA is separating the new part(s) from the original model file.

Status of Commercialization

The turbomachinery industry uses commercial grid generators (though not exclusively). This causes a problem for **CAPRI** specifically and automated analysis and design systems in general. The work here requires that the grid generator use the solid model during the meshing and update the information in **CAPRI** with the surface discretization when complete. Wrappers can be written to merge the operations but a more complete integration is desirable.

ICEM-CFD has signed the appropriate paperwork with MIT's Licensing Office so that they will be **CAPRI**'s commercial partner. ICEM-CFD will use **CAPRI** internally for their packages ICEM-Hexa and ICEM-Tetra (widely used within the turbomachinery industry). They will also provide external support for **CAPRI** for other organizations (though this may be problematic because some groups are in direct competition for the meshing marketplace). The goal for the near future will be to perform joint development. This obviously requires that they have and understand the source code. MIT will maintain the Intellectual Property rights.

An important goal with this commercialization is to provide free run-time licenses to all that have supported **CAPRI**'s development (which includes NPSS and the industry partners when using the NPSS software). Anyone that is not directly funding **CAPRI** will need to purchase support if required.

Status for Phase #1

- General reading support.

This is an on-going task to get the details correct for all of the geometry readers. In this project, getting all of the detail correct is of paramount importance. Only during extended usage do problems arise (usually when a part/assembly is loaded with some feature that is unusual). These problems continue to be addressed as they occur.

- Boolean Operator/Creation results.

CAPRI now supports Solid Boolean operators for all CAD systems. All ports (except for ProE) support the creation of simple solids. ProEngineer needs further work.

- Document the Geometry Viewer (GV).

The visual front-end needs will be documented by the end of this Phase so that others can use this software. It has been enormously valuable for the development and debugging of **CAPRI**. The GV software is stand-alone and could be used as a visual front end to any analysis suite. It has many useful features including 3D picking so that boundaries can be easily collected and displayed. There are versions for all of the supported workstations.

- Tessellation improvements.

CAPRI generates the discrete definition for the part for ProE and IDEAS. This is because the tessellation provided is not closed and conformal or there is no tessellation at all! This surface generation is an exceeding difficult task. The techniques used now are *ad hoc* and not satisfying. At times, the procedure fails leaving one or more faces without a tessellation. This has been addressed.

A new scheme for surface tessellation has been developed that is robust and contains the ability for the user to adjust the fidelity of the resultant mesh. It performs the triangulation a Face at a time by starting from the bounding Edge discretization. This insures that the result is water-tight.

- Writing & Design of the Master-Model Interface.

The Master-Model API enhancement has been specified. This can be seen in the attached document "Master-Model Interface Specification". The first expression of this API will be ported using SDRC's I-DEAS which should be in beta-test by the end of this Phase.

- Implementation of a Master-Model Design Example.

The Master-Model API will be used in concert with a grid generator and solver (currently NPARC's GMAN and WIND code) to produce an example of a complete **CAPRI** system of applications. A simple "one-off" turbine design code has been assembled using Parasolid. This work will become the test-bed for building systems that perform parametric studies and design optimization. This particular task will continue throughout the contract.