Mid-Level Cad Packages as Stepping Stones for Solid Modeling Education

Major Steven J. Schweitzer, Colonel John S. Klegka
United States Military Academy, West Point

Disclaimer: The views presented in this article are those of the authors and do not necessarily represent the views of the Department of the Defense or any of it's components. Also, the products mentioned in this article are not officially endorsed by the Department of Defense or any of its components.

Abstract

Introductory courses to 3-D solid modeling have the difficult tasks of both introducing 3-D solid modeling concepts and instructing the use of a particular software package to exercise these concepts. In many settings more time is spent learning how to use the solid modeling software than understanding the conceptual techniques of visualizing 3-D solid objects. The use of mid-level Computer Aided Design (CAD) packages with built in tutorials can greatly reduce the amount of time spent learning how to use the software and allow the instructor to focus on conceptual understanding.

Most mid-level CAD packages do not offer the power or functionality of the larger, more robust software used by industry, but their ease of use makes them a great first step in educating a student in 3-D solid modeling. Several mid-level CAD packages are made or supported by producers of large, industry accepted CAD packages. This relationship makes it easy to use the mid-level CAD package as an instructional tool toward teaching the industry standard CAD package. The instructor is still faced, however, with the task of selecting the appropriate software. This paper will describe a process for selecting an optimum package from among a variety of contenders. It will also explore the use of Parametric Technologies Corporation's (PTC) Pro/Desktop as a stepping-stone for the use of PTC's Pro/Engineer.

Introduction

In the past few years computers have begun to dominate the Engineering Design Graphic (EDG) curriculums. The hand drawing tools of the past are rapidly being discarded in favor of 3-D solid modeling software. A survey of EDG professional educators showed that over 40% of schools no longer teach the use of manual drawing equipment[1]. This fact implies that a heavy reliance is being placed on the use of computers for introductory instruction.   It also means that young engineering students are being required to master complex software packages while trying to grasp the fundamental concepts of engineering graphics.

The majority of EDG courses rely on high-end CAD packages for their instruction in 3-D Solid Modeling. Pro/Engineer is currently the most popular 3-D solid modeling package used for instruction[1]. High-end CAD packages are extremely powerful but they are also extremely

complex and difficult for new users to learn. This complexity forces the instructor to spend time teaching students how to use a particular software package instead of devoting the course lessons to conceptual techniques. Teaching through a piece of software is especially troubling given the turnover rate of software in today's fluctuating market. The end result is that both students and instructors focus on the use of software instead of the processes, techniques, and guidelines of modeling in 3-D. The recent emergence of mid-level CAD packages will help remedy this situation.

Mid-level CAD packages offer simple tools for creating 3-D objects with built in tutorials. Mid-level CAD packages sacrifice higher-level functionality but they are extremely easy to use. All educational departments that use CAD software should consider adopting a mid-level CAD package for undergraduate education. Some examples are AutoDesk's Inventor, PTC's Pro/Desktop, SolidWorks, SolidEdge, and IronCad. This paper will outline the process that the Department of Civil and Mechanical Engineering at the United States Military Academy at West Point followed for changing its CAD software and will present the results achieved with the new software.

Course objectives

The introduction of mid-level CAD packages will most likely occur in an introductory course. The mid-level CAD package will then offer the software foundation for all future CAD related courses. Thus the objectives of this introductory course should be tightly linked to the entire engineering curriculum.

Before considering any change in software the engineering program objectives should be reviewed and validated with respect to engineering design graphics and CAD. The course objectives for the introductory CAD course should be reviewed and revised to ensure that they support the engineering program objectives. The course objectives should be clearly written and be achievable. Avoid writing or accepting objectives that cannot be properly evaluated[2]. See *Mastering the Techniques of Teaching* by Joseph Lowman (pp 195-199) for further information on the creation of objectives.

The course objectives can begin with something as simple as ensuring all students can model simple objects in 3-D using a computer. Once the course objectives are set then the current software situation should be evaluated. Does the current software allow students to achieve all of the course objectives? If it does not, then a software change should be considered for the course.

Selection criteria

Once it is determined that a software change is needed, criteria for selection should be established. The selection criteria can be divided into two major sections: minimum requirements and evaluation criteria.

Minimum Requirements

The minimum requirements are items or capabilities that the software must possess. If the software does not meet these minimum requirements then it should not be considered. Below are some of the requirements that were considered in this study:

-The software can be installed and run on existing lab computers: Current lab systems should be analyzed to determine which of the candidate software will run on them. Most of the mid-level CAD systems are Microsoft Windows native software. The development of CAD software specifically for Windows has spurred an ease of use revolution. This ease of use is further enhanced by the fact the current student generation is extremely familiar with the Windows operating system (OS). If the lab is not equipped with Windows based PCs then it may also be time to consider upgrading the lab. This upgrade should not cause any problems with any legacy higher order CAD software as almost all of them now run under the Windows OS.

-The software imports needed legacy formats: The software should be compatible with any legacy software that will be in use. While it may not read the native file format, it must be able to import standard parasolid, ACIS, or IGES formats.

-The software exports required formats: The software should output formats that will be easily read by the higher-level CAD package in use. Also, if the students will use Rapid Prototyping or computer aided manufacturing (CAM) devices, the software must output a format that will be compatible with this process. (.stl, .dwg, .igs, etc.)

Evaluation Criteria

The individual mid-level CAD packages should be evaluated against a fixed set of criteria. The following criteria should be considered:

-Cost: This area is very complex. Spend time looking at all licensing options including using a server license. Investigate what happens if you decide to stop purchasing the software. Are you merely leasing the software or are you buying it? Get the vendors to price out similar license arrangements to accurately compare the costs. Does the cost include start up training for the instructors? Does the cost include the tutorials? Does the cost include comprehensive product support? If you cannot afford many licenses, does the vendor sell discounted student versions?

-Ease of Use: To get new engineering students engaged in CAD, the software should be as easy as possible to use. Easy to use software will reduce the amount of time spent teaching the software and leave more time to teach theory and concepts. The best way to evaluate this is to have an untrained person try to draw something with the software. If the process is easy and intuitive, the untrained individual should have little difficulty in making a simple object.

-Tutorials: Does the software have any tutorials available? Built-in tutorials tend to be easier (and cheaper) for the students but may not be as comprehensive. Do the tutorials merely teach point-n-click methods or do they attempt to impart creation theory?

-Computer Requirements: Does the software require "bleeding edge" technology? Will the current lab machines be able to handle large assemblies? Ensure that a large assembly is run on typical laboratory and student machines to check display performance. Many mid-level CAD packages require certain graphic cards.

-Both 2-D and 3-D creation: Ensure that all teaching options are available on the platform. Being able to create objects in either way gives instructors flexibility to meet the needs of individual student learning styles.

-Compatibility with high-end CAD systems: Will the software be able to work with the legacy software? Does the software link with any industry accepted high-end CAD package? If the mid-level CAD package works with a higher-end CAD package it will make it easier for students to transition to that package when required.

-Installation/License setup: This area is very important. How the software will be deployed in the labs and on the campus should be fully considered and evaluated. A great free program that is impossible to install will never be used. All options should be fully discussed with the system administrator who will be deploying the software. Software that requires any type of physical hardware tie (dongle, license based on system parameters, etc.) will cause much difficulty if the lab hardware is upgraded.

-Company's reputation: Never buy a piece of software from a company whose survival is problematical. If the software vendor closes their doors, you may have to repeat the software replacement cycle earlier than expected. Find out what other educational facilities are using the software and get their input. If no other educators are using the software, then an effort should be made to find out why.

-Product Support/Customer Support: What services are included with the software package? Are software updates free? How responsive is the company to customers? Call their technical support lines and see how long you have to wait.

-Integration with other Microsoft products: Some people may not like it but Microsoft is the dominant software environment. Is the software Microsoft certified? Evaluate how well objects can be moved through the Office package. Ensure that the software is Object Linking and Embedding (OLE) compliant. Remember that the students are probably extremely comfortable using Microsoft based operating systems and software.

-Animation capability: If a picture is worth a thousand words then an animation is worth millions. The software should allow students to quickly make some sort of animation.

-Motion Analysis capabilities: Does the software include any motion analysis?

-Compatibility with easy to use Finite Element Analysis software: Does the software link with or export to any of the finite element analysis (FEA) packages currently in use?

-Error analysis and repair functionality: Does the software come with any sort of healing functions to repair corrupted geometry? The more automated the repair system, the easier it will be on beginners.

A decision matrix or Pugh's method matrix should be derived using the above criteria. This matrix will provide a reliable method for objectively comparing all of the software packages[3]. The individual weights for each criterion will be highly dependant on the situation in the school/department. For many schools cost will perhaps be the most important factor. In other cases, capability may be most important.   A sample of the decision matrix utilized by the Department of Civil and Mechanical Engineering at the United States Military Academy at West Point is shown in Figure 1. For more information concerning the creation and usage of a decision matrix see *The Mechanical Design Process* by David Ullman, pp 160-164.

Criterion	Weight	Software Packages			
		I	II	III	IV
Cost					
Ease of Use					
Tutorials					
Computer Requirements					
2-D & 3-D Creation					
Compatibility with High-End Software					
Installation/License Setup					
Integration with Microsoft Product					
Animation Capability					
Motion Analysis					
Compatibility with FEA					
Error Analysis and repair					
Total +					
Total -					
Overall Total					
Overall Weighted Total					

Figure 1: Sample Decision Matrix for CAD Software

Selection process

Before beginning the selection process, a software selection team should be designated. This team should be comprised of representatives from every area of the engineering program that

will be affected. It may also be valuable to include a student who has already completed the introductory course with the old software.

The team or a team member should acquire demonstration versions of all the software to be reviewed. At least one person in the team should use the demonstration version to become moderately familiar with the software. Spending a short time with a piece of software should allow any CAD orientated person to discover its strength and weaknesses. This is also an excellent time to evaluate the tutorials for the software. Accurate time records should be kept while first learning the new software through its tutorials. Students will typically require three times the amount of time that an instructor takes. If it was difficult for a seasoned instructor to use the software then it might prove impossible for the students. Once the tutorials are complete, a list of question for the vendor should be derived.

Selected software vendors should be given the opportunity to conduct a full demonstration of their product at the school. Each demonstration should be held on a separate day. Each piece of software should be evaluated on its own merits and then the results compared. Having multiple vendors in one location simultaneously will ultimately end in direct comparison battles as each vendor tries to "win" the competition. The goal of this process is to make the students the winners.

It is critical that each vendor be given a typical laboratory and/or student computer to demonstrate how the software is installed. The computer that the vendor brings to the demonstration may be optimized to improve software performance. It is definitely worth the time and effort to have the vendor install and demonstrate the software from a typical lab computer.

Each vendor will invariably start with a canned demonstration. They will attempt to demonstrate the capabilities of their software with some complex, pre-drawn object. It is critical to have the vendor create something from scratch. If necessary, interrupt the canned demonstration. Ideally, each vendor should create the same type of object. The software sales people are often limited in their skill with the software, so creating an unfamiliar part from scratch can truly show how difficult a piece of software is to use. Try to avoid letting the salesman use a CAD expert to demonstrate the software. If the CAD expert seems to be dominating the creation process, ask to have him teach one of the selection team members how to create a 3-D object. By putting an untrained person at the keyboard it will be possible to evaluate how difficult the software will be to teach and learn.

When the demonstration is finished, each selection team member should complete the evaluation items in the decision matrix for that vendor. The only exception could be the cost. It is best to discuss the price of the software in a closed forum. Each vendor will have educational pricing incentives. Ensure that you fully investigate all pricing schemes. Prices will also vary on how the license will be implemented. A server license option may be slightly more expensive but it will allow students to run the software from any location.

When all of the vendor demonstrations are complete, the decision matrix should reveal a clear choice. It is important not to get bogged down in the individual slight variations in each piece of

software. Each piece of software is sure to have some unique and interesting features but stay focused on the criteria. A nifty widget that is unique to a particular piece of software may be more of a hindrance in the long run, especially if a multitude of different CAD software is being utilized.

Results

West Point's Department of Civil & Mechanical Engineering followed the above outlined process during its CAD software selection. After receiving demonstrations of four mid-level CAD packages, West Point decided on PTC's Pro/Desktop. The demonstrations proved that the mid-level CAD packages functionally performed on the same level. Each package had minor differences and varying levels of advanced functionality but, at the beginner creation level, they were all very similar.

Pro/Desktop's major advantages in the Evaluation Criteria were in ease of installation, depth of built in tutorials, cost, and direct compatibility with its big brother, Pro/Engineer. The complete compatibility with Pro/Engineer and many of the other Pro product line meant that Pro/Desktop could be used as a training tool for students who will most likely use Pro/Engineer in industry. Of more concern for this selection process, however, is the ability for students to work with Pro/Engineer files that are available from Capstone Design customers during their final capstone design experience as undergraduates. Autodesk Inventor also now shares this same functionality with Autodesk Mechanical Desktop. At the time of our review, Inventor was only in Beta release.

West Point could buy enough licenses for all Mechanical Engineering students to run it on their individual PCs. Being able to do this was considered critical since our students were more likely to learn and use the software if they didn't have to travel to a computer lab to use it. This also meant that the system requirements had to play a crucial role, as most students do not have cutting edge technology in their rooms.

If you are already a PTC customer you probably already have Pro/Desktop. PTC markets Pro/Desktop as a conceptual design package for everyday engineers[4]. It typically comes bundled with other PTC programs.

Use of PTC's Pro/Desktop

In the fall of 2000 West Point moved from Pro/Engineer to Pro/Desktop as its tool for introducing 3-D solid modeling. Solid Modeling concepts were taught using Pro/Desktop in eight, 55-minute classes and two, 2-hour labs. The built-in Pro/Desktop tutorials were the primary instructional source supplemented with two instructor-designed tutorials. The students completed basic tutorials, advanced tutorials, and the command and technique exercises. They also completed a gear lab (Figure 2) and a pin connector lab (Figure 3). The ten EDG lessons culminated with a Remote Control Airplane design.
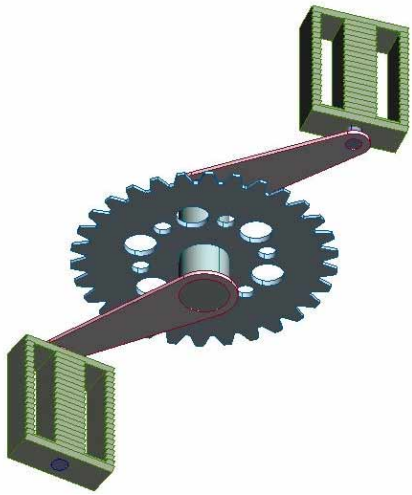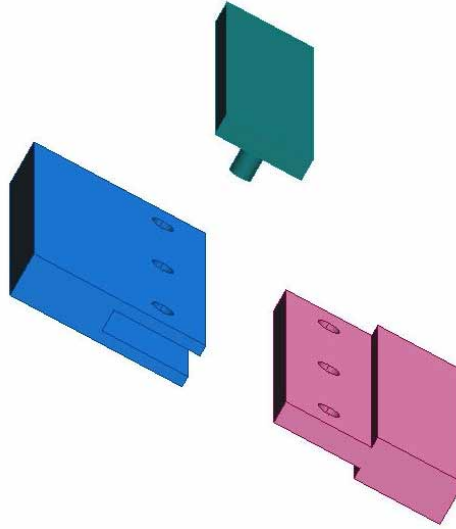
Figure 2: Gear Lab   Figure 3: Connector Lab

Rapid Prototyping was also taught and conducted using Pro/Desktop. The only requirement for rapid prototyping is for the software to export a standard stereo lithography file (.stl). Moving files from Pro/Desktop to the Rapid Prototyping printer was extremely quick and easy.

Finite Element Analysis (FEA) was conducted by paring COSMOS/M DesignStar with Pro/Desktop. DesignStar is an easy to use FEA program that imports a wide variety of geometries. The screen layout of DesignStar bears a striking resemblance to Pro/Desktop and many other mid-level CAD packages. Moving models from Pro/Desktop to DesignStar was very easy but Pro/Desktop lacks direct integration with any FEA package. This lack of integration makes it difficult to conduct optimization studies.

Assessment plan and results

To assess the implementation of Pro/Desktop a time analysis survey was conducted for comparison to the previous semester when Pro/Engineer was taught. The students recorded the number of minutes it took them to complete each assignment and to prepare for the next lesson. An individual survey was also completed at the end of the EDG instruction to determine how comfortable each student felt with 3-D solid modeling.

The Pro/Engineering final project was to draw and assemble a nose for an Army missile system. Students were provided with all other parts in the assembly (Figure 4). The Pro/Desktop final project was to draw and assemble a nose cone, a tail cover, and a wing for a remote control airplane. (They were supplied the fuselage and the engine, see Figure 5) The airplane project drawn on Pro/Desktop was approximately 3 times harder but the students spent only 1/2 of the time the Pro/Engineer students spent on the missile nose cone. The students using Pro/Engineer spent an average of 250 minutes while the students using Pro/Desktop averaged only 125 minutes. Figures 5 & 6 show a sample of a student turned in project for the RC Airplane assignment.
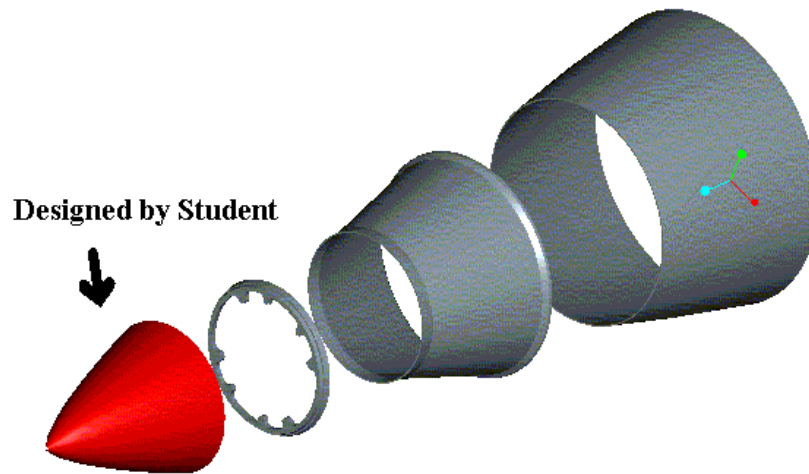
Designed by Student

Figure 4: Army Tactical Missile Nose Cone drawn with Pro/Engineer
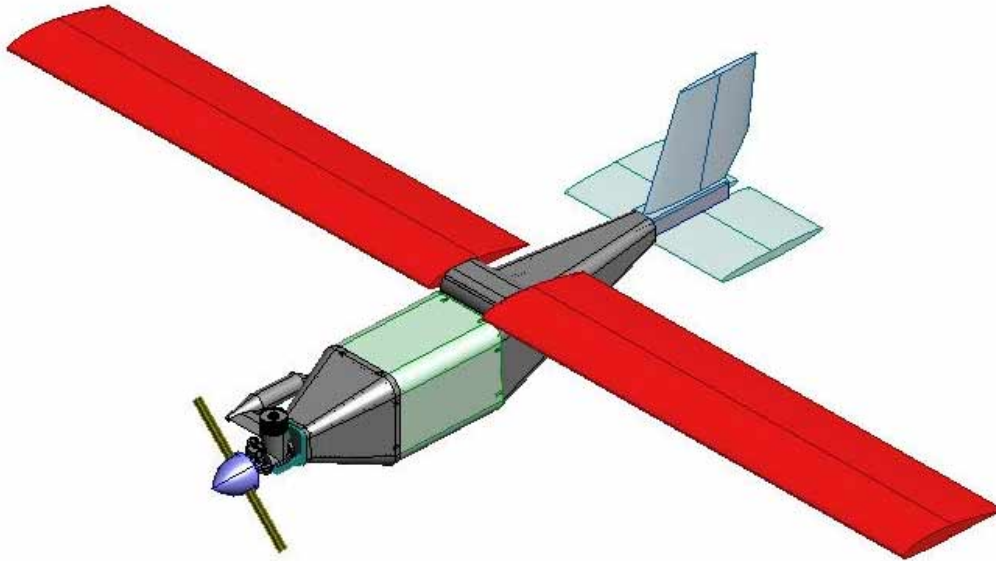

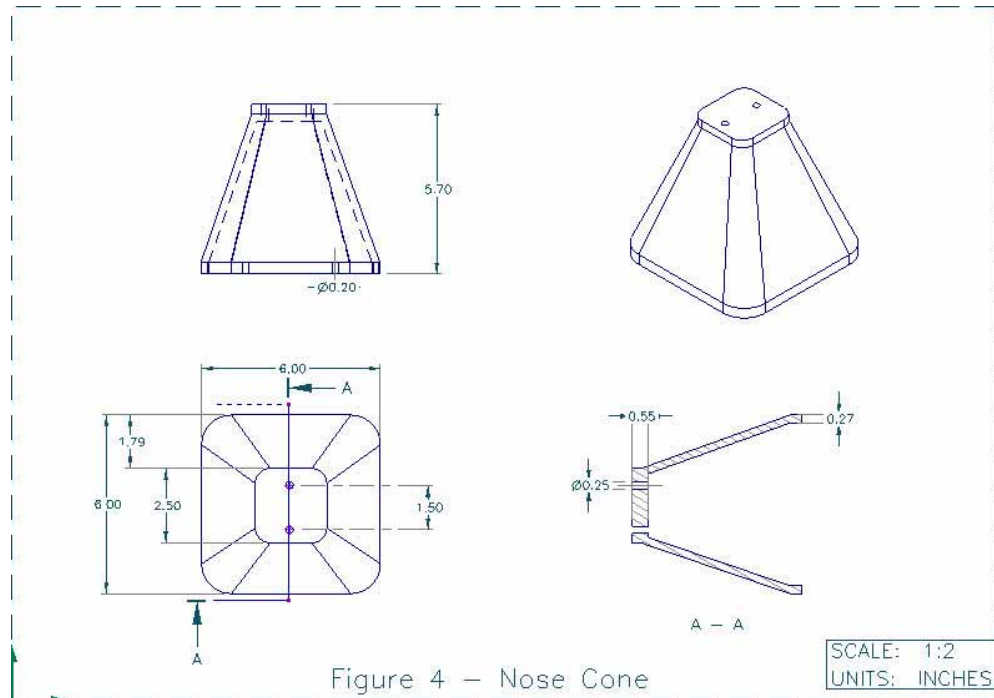Figure 5: Remote Control Airplane Concept drawn by Students with Pro/Desktop

Figure 6: 2-D Drawing of Designed Nose Cone produced from Pro/Desktop 3-D Model

The student's responses on the individual surveys also showed that after 10 lesson students felt much more comfortable with Pro/Desktop. It also suggests that the students are more likely to use CAD in their future classes. This will have to be examined as the student progress through the curriculum. A sample of the survey results is shown in Figure 7.
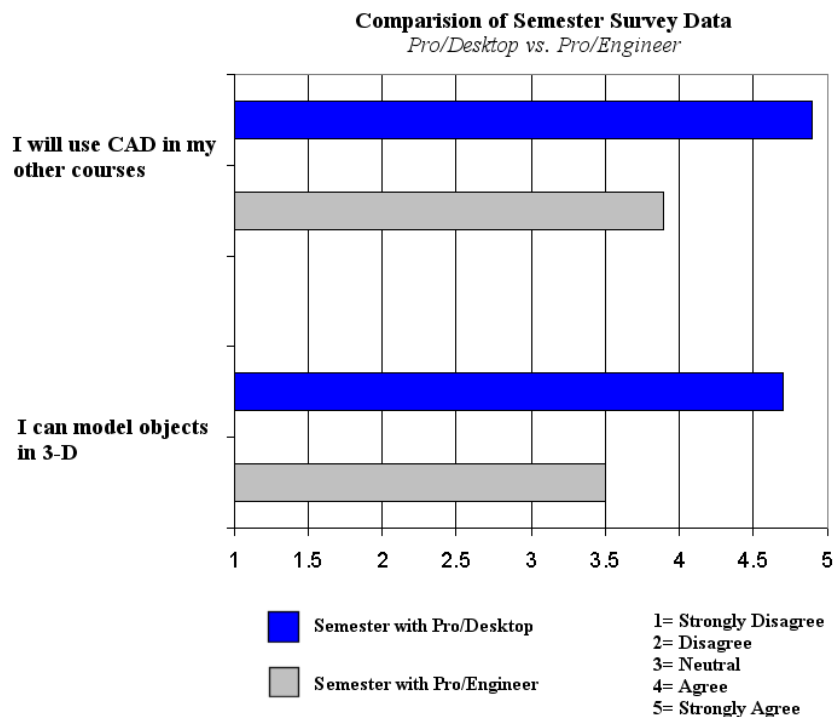

Figure7: Comparison of Student Survey Data.

Further assessment is in progress through the senior level capstone projects. The seniors in their engineering capstone courses received no formal training with Pro/Desktop. Their mid-year briefing showed on high reliance on Pro/Desktop to layout their prototype vehicles. Project team members essentially taught themselves how to use Pro/Desktop with help from students currently in the introductory class using Pro/Desktop. The level of details in the CAD drawing this year is far greater than any previous years. Unlike previous years, the students are actually using CAD in the concept phase of the design process. This clearly demonstrates that mid-level CAD packages are an effective tool for integrating CAD into every student's design work. Figures 8-11 show capstone project results of students who received formal Pro/Engineer training but taught themselves Pro/Desktop after the software change was made.
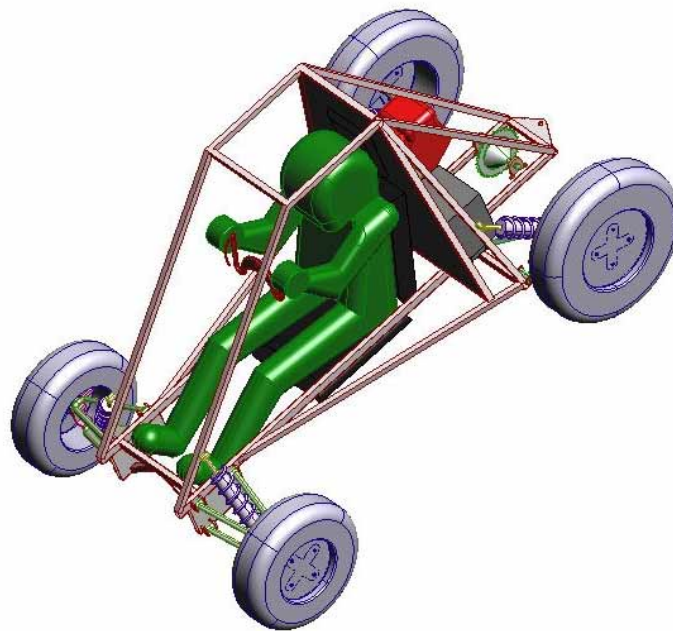

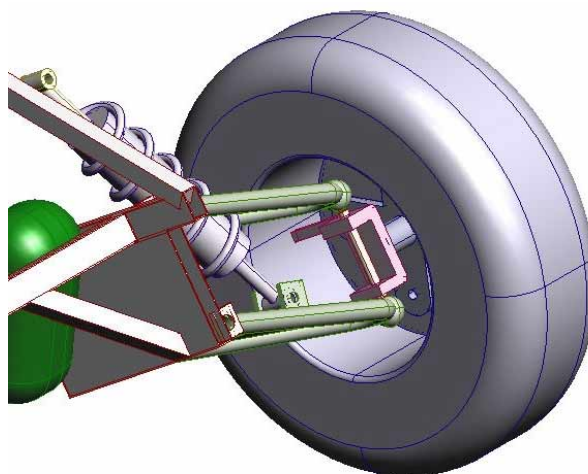Figure 8: West Point Mini Baja 2001 Concept

Figure 9: Close up of front left Mini-Baja Suspension

Conclusions

Mid-level CAD packages can drastically reduce the program learning time for undergraduate students. This gives instructors more time to address conceptual issues, which in the end produces students that are better 3-D solid modelers. Pro/Desktop has proven extremely successful at the United States Military Academy. Pro/Desktop like most mid-level CAD packages lacks many of the advanced add-on features of packages like Pro/Engineer or CATIA. However, at an undergraduate level these mid-level CAD packages give students more than enough tools to create useful 3-D solid models. For general engineering students, the mid-level CAD packages offer a conceptual creation solution with minimal software learning time.

These mid-level CAD packages, while easy to use, expose students to creation exercises that are very similar to skills needed for using high-end CAD packages. This makes them ideal for use as an intermediate step in professional CAD instruction.

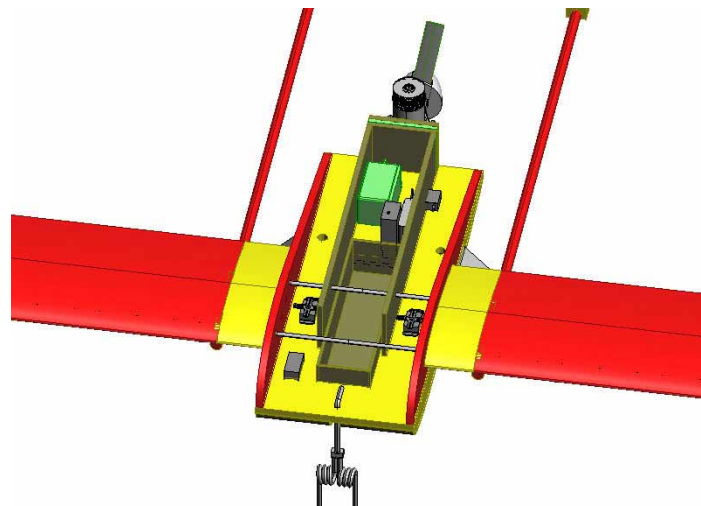

Figure 10: SAE Aero 2001 Concept

Figure 11: SAE Aero 2001 Fuselage Interior

Bibliography
1. Clark, A & Scales, A. "A Barometer for Engineering and Technical Graphics Education", Paper presented at the ASEE 2000 Conference, Session 2438.
2. Lowman, Joseph *Mastering the Techniques of Teaching*, Jossey-Bass Publishers, San Francisco, CA 1995.
3. Ullman, David, *The Mechanical Design Process*, McGraw Hill,Boston, Massachusetts 1997.
4. White Paper, Conceptual Engineering, http://www.ptc.com/products/desktop/concept_eng.pdf

Mid-Level Cad Software

PTC Pro/Desktop: http://www.ptc.com/products/desktop/index.htm
SolidWorks: http://www.solidworks.com/
Iron Cad: http://www.ironcad.com/
Autodesk Inventor: http://www.autodesk.com
SolidEdge: http://www.solidedge.com/

COLONEL JOHN S. KLEGKA
Colonel John S. Klegka is the Director of the Mechanical Engineering Program in the Department of Civil and Mechanical Engineering at the US Military Academy. He has had a variety of command and staff assignments in the United States, Germany, and Korea. He received the BS degree from the US Military Academy, the MS degree from the University of Michigan, and the PhD degree from Texas A&M University in Mechanical Engineering.

MAJOR STEVEN SCHWEITZER
Major Steven Schweitzer is the Course Director for the Computer Aided Design courses at the United States Military Academy. He also helps instruct several Aerodynamic Courses by piloting the students in required flight labs   He has a commercial instrumented license in both a fixed wing and helicopters. Steve is a Captain in the United States Army. He has served in a variety of assignments including a Company Command at Ft. Eustis, VA. Steve received a B.S degree in Mechanical Engineering from the United States Military Academy in 1990 and a M.S. in Aerospace Engineering from the Pennsylvania State University in 1999.

Disclaimer: The views presented in this article are those of the authors and do not necessarily represent the views of the Department of the Defense or any of it's components. Also, the products mentioned in this article are not officially endorsed by the Department of Defense or any of its components.