FUTURE OF COMMUNICATING DIGITAL DESIGN IN ARCHITECTURE

Overcoming the Divisive Power of Computer Aided Design

FLORIAN TECHEL
Department of Architectural Engineering, University of Sharjah, Sharjah, United Arab Emirates
techel@sharjah.ac.ae

Abstract. A few decades ago architects, engineers and the building industry relied on a set of self-developed tools for drawing and standards for communication within the profession and beyond. Everyone involved in the process of building understood these standards that were developed, controlled and updated by the profession. Today the situation appears more ambiguous. The introduction of Digital Media, and specifically Computer Aided Design, has greatly enhanced the potential for productivity gains. On the other hand, the lack of standardized open file exchange formats in CAD has created communication barriers by making data exchange more confusing and ambiguous. Frequently this has consumed the very productivity gains that were originally envisioned by industry. Problems with proper and fluent data exchange between software applications to no small extent are due to fundamental disagreements between software designers on the proper digital description of a building, leading to nearly insurmountable communication obstacles, designed to potentially divide the profession, practitioners and the educational environment. Consequently construction has not partaken in the productivity gains that other industries have enjoyed. Proprietary file formats and closed software systems have fostered the development of design camps that rally behind one software. Others reluctantly buy into certain "solutions" for they are perceived to be *standards*.

Innovation is hampered as development of industry design tools is no longer controlled by architects, engineers and the construction sector but instead by private software companies frequently pursuing their Based on 20 years of experience with CAD in the profession and academia this paper critically investigates the status quo of CAD in the building industry. It points towards strategies of overcoming the current problematic situation and putting the profession back in control of its own communication process.

1. Ramifications of CAD-use on the Professional Practice of Architects

1.2 HISTORIC BACKGROUND

"The technical drawing has developed over a considerable period of time, with the goal of delivering clear instructions for the assembly or construction of an object (Robin Baker, 1993)."

In the "paper age" architects and all the parties involved in the building process, had control over the process of designing and sharing the design with the other parties involved. Countless rules for communication had been developed over the process of hundreds of years, some of them by the profession as a whole; some of them were developed locally between building participants in different regions or countries. All of these rules had basically one goal in mind: to facilitate the communication about the object of interest, *the building*.

The profession itself controlled the rules of communication, and therefore it was relatively easy for the people involved to develop rules and systems of passing along this knowledge from one generation to the next. Different architecture schools have tried (sometimes radically) different methods of teaching this knowledge, but eventually a roughly similar canon of skills and knowledge was passed along from one generation to the next. Changes were implemented in an evolutionary fashion.

These skills involved the use of *real tools* such as pencils, colored pencils, rapidographs, rulers, T-squares, parallel bars, stencils, etc. Each and every one of these tools usually served only one task or a very limited amount of tasks, but that ability was usually almost self-explanatory by simply looking at the tool. In those cases that the tool was not self-explanatory, one usually had to watch someone else use the tool and then could pick up its inner working within a few minutes. The tools themselves were so neutral that they permitted the cast of architects to develop its own working rules and styles.

This is not to say that mastering these tools was easy, in fact whole semesters were (and still are) spent in the schools trying to teach these representational skills. But the mastery over these tools did allow the user a certain freedom to develop an individual style out of the core skills. The knowledge over this core of skills made professionals qualified enough to be compatible (and employable) across vastly different offices.

It was therefore common practice that in a job interview the office manager would ask: "can you draw?" and when the answer was affirmative (as it usually was) the person in question was hired for a limited time-span, after which there usually was an evaluation just how good that person "could draw" and what his other skills were.

1.2 CURRENT STATUS

This situation has changed dramatically with the introduction of computers in general and CAD systems in architecture in specific. The introduction of digital media was a revolutionary change to a profession in which representation had not changed substantially within hundreds of years. Disengaging the medium (the CAD file) from its representation (the printout) has radically changed the way in which architecture is drawn. Within a brief transition time the question of "can you draw" was replaced with "do you know how to operate a CAD system?" (general question) and if the answer was affirmative, the candidate was hired. But the older architects, who mostly purchased CAD systems not out of an inner determination, but for alleged market forces (everybody else had one and they did not want to fall behind) started to gather that there are different CAD systems, while the young architects knew that already out of school.

Before long, the question of "do you do CAD" was therefore replaced by the current question: "can you operate the '*enter certain brand name here*' CAD system?" Often enough the answer to this question is negative, and that forecloses any possibility of employment in that very architectural practice. This gives rise to the legitimate question of: does the profession still control its tools or do the tools start to control the profession?

1.3 RAMIFICATIONS OF THE STATUS QUO ON THE PROFESSION

This qualification to operate whatever specific "market leading" CAD-application has started to dominate other qualifications that comprise an architect. That trend may be considered dangerous, for several reasons. First of all software applications change their version numbers quicker than architectural trends (which already change at a breathtaking pace) to the point that an architect has to spend a significant amount of time simply trying to stay current in the field of software. This is especially strange since the basic *principles of CAD* have not changed very much over the past decade.

The situation has deteriorated to a point that the mastery over one specific CAD application, or a core suite of software titles, has often become the primary reason for employment. This neglects or belittles the many other skills and knowledge that any (prospective) architect has acquired throughout his career and education and potentially reduces the young architect to a mere computer operator, just that he/she does not operate mainframe terminals, but a CAD-workstation.

1.3.1 Trivializing Knowledge

The reduction of the knowledge of an architect primarily to his capabilities of operating a specific suite of CAD-programs, modeling and rendering applications, trivializes the body of knowledge that has accumulated within the profession over decades, if not centuries. It potentially devalues the standing within society and self-esteem of an entire profession.

Ultimately this will have an effect on the institutions that harbor this body of knowledge, the schools of architecture, and the educators working within. How shall the educators of CAD maintain their status within two systems (that one of architecture and that one of educators) if potential students start to question a five year long architectural education costing them (or in most cases: their parents) large five-figure or six-figure amounts for their architectural education? What if the sentiment starts to dominate that their chance of employment would be infinitely better by taking a six-month long crash-course in “abc-CAD?”

This trend could be enhanced by the (somewhat heretic) comparison of what the three possibly most influential architects of the 20th century (Le Corbusier, Mies van der Rohe and Wright) have in common? They all lacked a formal architectural education. Has this harmed their careers? This anti-academic trend is being fostered by happenings in other fields, primarily the booming computer and software industry. The role models of the information age, the Gates’, Jobs’ and Dell’s all share being university dropouts. Has this harmed their careers? Why should someone invest hundreds of thousands of dollars into the education from a highly recognized university? With the same money, he/she could found an upstart company on one of the many “insanely great” ideas floating around in cyberspace, take the company public after two or three years and enjoy oneself for the rest of the life. With all the money made during that brief time, one can buy all the architects and/or lawyers in the world.

This trend could be called the “cannibalization of education.” A set of values that has developed within the bourgeoisie over the past two centuries, namely that “a good education” is the primary insurance against poverty and towards a migration into the upper ranks of society, possibly even the highest ranks. This understanding is being gutted in light of the “revolution of information technology.”

In the field of architecture, the vision of capital-intense “CAD-sweatshops,” where countless numbers of architectural drones are working in droves to finish their individual chunks of drawing tasks is not entirely far-fetched. What may remain are a few (and far between) star architects, the pop-icons of an “educated” elite, and countless nameless drawing clones which spent way too much money and time on their education.

This would be a return to times of the past, when the emperors only held a handful of architects, and the overwhelming mass of buildings were designed and constructed by masters of the building trade (either masons or carpenters). It is at this point in time that the profession of architects has to decide where it wants to head from here, if it wants to lead, follow or get out of the way.

1.4 AN ANALYSIS OF THE UNDERLYING PROBLEMS

1.4.1 How did it get this way?

The computer is an odd tool, in that it claims to being able to do just about anything one could possibly imagine. It converts itself from a number cruncher into a game machine, into a stereo system into a communicational device into whatever-else-you-can-think-of by merely launching a new application. It is therefore difficult, if not impossible, to develop a correlation between what the computer does, and how it should look, or how it should work, as there is no immediate correlation. There is nothing imminent in a computer as is in a compass or a pencil.

It is therefore possibly not proper to call a computer a tool; it is more of a *container for tools* (the software applications). This jack-of-all-trades image has unfortunately colored off to the software itself, the individual applications. These applications are the objects inside the computer, which most likely resemble traditional tools. Unfortunately this opportunity was soon missed by the industry as the makers of software tried to stuff as many functions into their applications as possible. What usually suffers is the functionality. Purists still claim that the original versions of the Apple Macintosh applications, MacWrite and MacPaint, are still unrivaled in their immediateness and understandability. Some people claim that new Apple applications such iTunes, iPhoto, iMovie, etc. have regained some of that original spirit. To cite Ludwig Mies van der Rohe: “Less is more!” This is what good tools should be all about, no matter if they are real drawing tools, or computer tools.

Floating palettes are only the most recent admission of software makers, that they themselves are increasingly at a loss *just where* to put all these functions on the computer screen. That people actually have to memorize these functions, and work with them, appears to have passed them by. This is a classic case of the mix-up of quantity and quality, an example of bigger not automatically being better.

1.4.2 Different ways of working

Computers developed as *application centric* machines, which is usually very different from the *document centric* way in which humans normally operate. Especially architects traditionally would span a document onto the drawing table and then place all the tools (pencils, rapidographs, rulers, compasses) on that drawing or in its immediate proximity. The document was the very center of attention, sometimes for weeks and months. There are many reasons why this document-centric approach is far more comprehensible to the human mind. And whether it is a writer, a photographer or an architect, they all usually deal with documents and all their traditional tools (typewriter, pen, camera, dark room, drafting table technical pens) all gravitated around the creation of these documents.

In the original days of computing, this lack of document-centeredness was justified by the lack of power of early PCs.

Current computers are approximately 1000 times as powerful (calculation power, volatile and permanent memory) as PCs in 1980, when the first mass market CAD applications were developed. The question may therefore very well be posed *just when* computers will be powerful enough to challenge the traditional application centric ways of computing?

As it turns out the software industry itself has little if any interest in changing this status of application centric computing. For one, CAD applications are among the most complex software applications and many companies simply shy away from rewriting millions of lines of code. On the other side, CAD vendors have found that the application centeredness is the easiest way of binding clients to the own software (see Figure 1). Making computer users depend on a single software application, the alleged *standard*, has enabled the development and solidified the power of software empires.

Wikipedia defines: “As each CAD system has its own method of describing geometry, both mathematically and structurally, there is always some loss of information when translating data from one CAD data format to another. The intermediate file formats are also limited in what they can describe and can be interpreted differently by the sending and receiving system.” It is troubling that this definition simply accepts this problematic as a status quo.

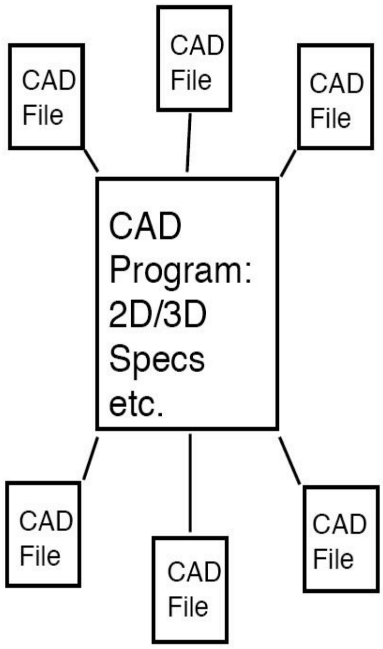

Figure 1: In an application-centric model the exchange of data is limited to a specific software, at times even a specific version of the software

This not only happened in the field of CAD, but in Word Processing, Spreadsheet Applications, Databases, Graphics, etc. In these fields, however, in most cases it is possible to move information back and forth in a loss-less way, by way of neutral file formats which are supported by an increasing number of software vendors, if only with grinding teeth. Some of them still make every effort to introduce "features" into their software that will not interface correctly with the neutral exchange standards and therefore render it useless once a substantial amount of users uses this software. The Open Source Software movement has created additional momentum towards the free flow of information by use of open file-exchange standards. Free-of-charge applications such as OpenOffice (office application set), Firefox (webbrowser), Apache (web server), SQL (databases), clearly started to challenge their respective proprietary market leaders.

On the other hand, the file content of all the applications above is of a substantially simpler structure. Text documents, Spreadsheets and Databases, for example are all based on pure ASCII-text, while pixel-based graphic formats contain binary information, but of a comparatively simple structure. CAD data is dramatically more complex and consequently CAD files have a drastically higher level of internal complexity.

1.4.3 Establishing Empires

Over the past decade or two we could witness the establishment of such overwhelming organizations as the “AutoCAD Nation,” the “MicroStation Republic,” the “Empire of Allplan,” the “Land of ArchiCAD,” the “Dukedom of DataCAD,” etc. The largest one of them wields the biggest sales argument in its own favor: the alleged *compatibility* for few of them really have an interest in facilitating data exchange with other applications. This ties the clients to the vendors (compare Figure 2).

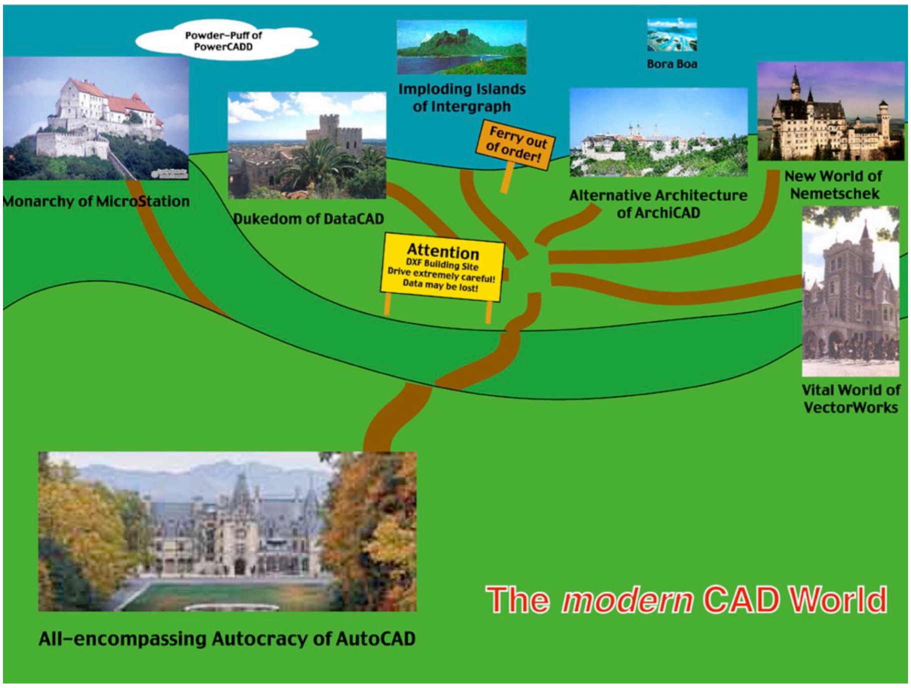

Figure 2: A view of the data exchange between selected mainstream CAD-applications

As most users need to communicate their designs, file compatibility all of a sudden becomes the facilitator of mere survival in the digital world.

There was a time when the trade groups themselves (the architects, the engineers, etc.) decided over their own drawing tools, over their own drawing rules and subsequently over the rules for information exchange. In general, each and every participant in the process couldn’t care less, *how* the drawing was created, and *what tools were used* in the process, for as long as the final document spoke for itself.

Today many of these standards developed by profession, are in vain, as the software in use generally lacks any kind of intelligence and does not understand the simplest drawing if it originates from another application. At times the applications in use do not even comprehend their own data output.

What good are standards for printed copy when an increasing amount of documents is being exchanged in electronic form?

Where does this leave the profession of architects, and all those involved in the building process?

1.4.4 Exchanging Data

AutoDesk, the maker of AutoCAD, by its own market dominance developed and pushed the data-exchange-format (DXF), and later AutoCAD's native file format (DWG), as file exchange standards into the market. Due to AutoDesk's power every one else had to follow their conventions or face a tiny market niche. In the process the building trade groups surrendered most of the authority over information exchange to the maker of one CAD application. The CAD-data-exchange standards proposed by the American Institute of Architects are nothing but a reflection of the (limited) abilities of AutoCAD.

Especially for the field of building, the purely geometric approach of DXF and DWG leave many things to be desired. While architecture and building construction play a small role in the overall CAD software market, problems faced in the building industry, may be by an order of magnitude more complex compared to the abilities that normal CAD applications usually show. Most instances of building construction produce small, if not smallest, production runs, and therefore usually do not warrant the design of complex geometries, and the necessary molds. Consequently building geometries are usually simpler and Architectural CAD systems need not be as *geometrically* complex.

1.4.5 Architecture is Different

While most people have a pretty good idea about the overall looks and proportions of a mainline brand name car, only few, even architects, are able to comprehend even approximately the relationship of all the parts and components of a medium sized building. This is because (good) architecture is *spatially* complex.

Let this be illustrated by the example of a simple doorknob. That knob belongs to the door, which itself belongs to a frame, which is installed in a wall, between two rooms, on a floor in a certain wing of the building. These are already six levels of dependencies, something that even most of the specialized Computer Aided *Architectural* Design applications cannot deal with today. Another big problem is that buildings tend to be relatively repetitive from one floor to the next. It therefore makes a lot of sense to organize drawings into different *classes of objects* on the one side (walls, columns, slabs, openings, etc.) and the *individual floors* on the other side. Today only the fewest of advertised CAAD applications support this need. Most CAD applications do not support organization of objects beyond the ability of using layers.

To sum it up, architects and other planners currently face:

- Software that is so complex that it often requires months to grasp the core working of the software.
- Individual CAD-applications, although they all do similar work, often do it in a vastly different way. Knowledge and skills acquired in one application are of a substantial smaller value when using a different application.
- A data exchange format which lends itself to the description of two-dimensional objects, but falls short when it comes to the true *volumetric* description of 3D-objects and their behavior (e.g. structural and thermal).
- Data-exchange format lacks proper dimension support. That means it does not know if the objects drawn in it are dimensioned in meters, feet, apples or oranges.
- Software that is difficult to scale by the user, if at all. Scaling means the ability to purchase the software according to ones own needs and not in the increments prescribed by the respective vendor. This means the ability to purchase individual components, which are able to interact with one another.
- Unnecessary and undesirable dependencies to CAD-applications and the software vendors that make them.
- Innovation cycles over which the users have little, if any, control. The core of most of the CAD software in use today is some 15 to 20 years old, only has been repackaged several times since with incremental innovations at the discretion of the vendor *not the clients*.
- Ideological debates, not over the design content, but the politically correct CAD software to use.

2. An Outlook

It is not the intention of this paper to devaluate the potential benefits of CAD for architects. CAD, used correctly, may be one of the biggest liberators of redundant and repetitive work and will allow the creative designer to concentrate on the true issues at hand. Such systems may continuously and immediately provide the designer with the necessary information to make competent design decisions. These are not restricted to spatial or aesthetic decisions but could involve, for example, thermal simulations where design decisions could be immediately effected by their energy impact, especially with high energy prices.

Currently these interactive schemes do not work in most cases, not because of a lack of power of the computer hardware, but mainly because of the deficient software. The way in which most CAD systems describe the building is, at best, an arcane graphical representation, and not a holistic volumetric description.

It also does not work, because most CAD-applications are closed-off worlds of their own that do not interface or interact with other applications.

In the view of the author it is high time for the profession of architects, but especially for the academicians who (should) work on the forefront of CAD education, to think about how architects can emancipate the profession from the dominance of the software makers. This would be the first step towards regaining the control over the very own communication tools of the profession.

The dominance of the CAD industry over the modes of communication can only be overcome by individual professional bodies starting to recognize the problems and develop actions.

2.1 ACTIONS

While the average CAD users currently spend a fair amount of time disputing which is the alleged best CAD-software, the target is increasingly lost out of sight: architecture and the process of building itself. Would it not be great if the discussions over the alleged best CAD-software would be a matter of the past? Would it not be great if the exchange of building data would simply *work* independent from the used software?

2.1.1 A Call for a Different File Format

Currently the biggest stumbling block towards a greater freedom of data exchange is a file format as neutral as the traditional architectural paper plan.

The dominating DXF/DWG-format have one advantage and several deficiencies. The advantage lies in the fact that it is in many cases a least common denominator. One of the disadvantages is that it follows a simplistic model metaphor. Why should DXF/DWG support the description of volumes (something that architects are usually interested in the moment it comes to specifying the quantities in the plan for bidding) when the mother application (AutoCAD) is still mainly focused on the description of complex surfaces? For as long as CAD is utilized primarily in a two-dimensional fashion and only interchanges the 2D-plans of the building, DXF/DWG works relatively proper, for as long as sender and recipient of the file are aware of the restrictions imposed by limitations of the underlying file format.

The biggest obstacle is, however, that DXF/DWG is *not* a standard. It belongs to a privately held company that follows its own interests. These interests lie, due to mere numbers of licenses sold, outside of the field of building. It is therefore no surprise that the company does not support the building sector and its metaphors in a desirable fashion. It is primarily marketing that tries to convince architects and civil engineers that what is good for the electrical engineers, and the aeronautical engineers and the mechanical engineers, simply *has* to be good for the building industry as well! A carpenter would have to be able to do good work with a mason's hammer. After all, it has been tested millions of times with masons, so there should be no reason that the carpenter should have any problems.

Every time that AutoDesk introduces a new version of its market leading CAD-application, it also introduces a new version of DXF/DWG, which usually raises backward compatibility issues. Then all the other vendors play catch-up for the following year in order to make their programs compatible once again.

Standards, on the other hand, are so prevalent in just about every single aspect of our lives that we all but forget about them. From the cement in the foundations of our houses, to the quality and the dimensions of the wood, to the dimensions of the pipes, to the color and durability of the paint on the façades, just about everything we use in the building industry follows established standards. There is simply no reason why this should not be the case for the very documents we produce to properly document the building as a whole. These standards do exist, for as long as the drawings are done on paper. By comparison there are preciously few standards for digital CAD-documents in the building industry? Most of them were developed within the constraints of the dominating CAD application, AutoCAD.

As the industry transitions to an all-digital planning process it would appear normal that digital standards emerge that are as specific to the building industry as the paper standards used to be.

There are few independent bodies, which oversees the proper definition and implementation of a data exchange standard in the CAD field and not one of them is fully acknowledged by any professional organization such as the American Institute of Architects, the Royal Institute of British Architects or the German Chamber of Architects, to name just a few. While CAD files are dramatically more complex than web pages, there is no reason why data exchange should remain the proprietary privilege of one or a few single software vendors. Open file formats have been proven to work in the field of word processing using RTF, computer graphics in the form of JPEG, TIFF, PNG, etc, with databases in the form of SQL, spreadsheets utilizing SYLK, and several other areas. The very existence of the World Wide Web is based on the adoption and use of public open file formats. Consequently it is possible for the average user to use the web focusing on the content rather than the technology behind it.

In order to facilitate an open system of free flowing information between various applications of differing vendors, the development of a stringent, yet flexible data format for the comprehensive description of buildings should be a prime objective. There have been efforts in this direction since the mid-1980s that have lead to the development of the "Standard for the Exchange of Product Model Data," or better known as STEP (STEP ISO 10303). This is a whole family of data format standards. The part of STEP that is of the biggest interest to architects is ISO 10303-225 "Building Elements Using Explicit Shape Representation." More recent standardizing efforts to facilitate file exchange between CAAD software lead to the standard ISO 16739 (About SC4 Standards). These "Industry Foundation Classes (IFC) are a standardization initiative by the International Alliance for Interoperability (IAI) to produce an open standard that properly describes buildings as a set or system of interrelated objects and components that all have specific properties, such as dimensions, relations (to other objects and/or components), costs, thermal conductivity, thermal mass, etc. Important is the fact that IFC is conceived as an *open* standard that can be expanded and developed. The most recent version is IFC 2x2 (July 2004). Software vendors may write their own interfaces to support IFC and may request verification and subsequently label themselves as IFC-certified. Several applications have requested the verification procedure (List of verified applications). Overall the initiative tries to include a much broader set of information into the description of the building, an effort that will lead to more precise planning through digital simulation before construction. This could be called *smarter planning and building,* consequently it is also referred to as "buildingsmart" (Description of Buildingsmart).

Addition: The author is delighted to find that some momentum is gathering behind the IFC standardization efforts. After submission of this paper, several articles were published on the website of the American Institute of Architects (AIA). One article about the development of the US National CAD standard (NCS) makes specific reference to the IFC format and the International Alliance for Interoperability (Tardif). Another article by Dianne Davis, president of AEC Infosystems explains that "BIM [Building Information Modeling] is what we all expected computer-aided design (CAD) to be…" and Glenn W. Birx, AIA explains in his article of December 2005, how the introduction of the BIM concept changed the work culture in his company.

Also, at the upcoming BuildingSmart Conference in April 2006 in Munich, Germany a representative from the US General Service Administration will report on their experience with IFC.

2.1.2 More open software

Once this mutually beneficial and inherently flexible data format is in place and supported by all relevant forces in the building industry (planners, administrations, contractors, material manufacturers, etc.) the dependency of the industry to proprietary software is greatly reduced. No longer is the possession of one specific piece of software required for compatibility and the legitimate communication requirements of the industry. Instead all market participants may opt to purchase software of their own liking for as long as it is certified to support the open standard.

Consequently the industry may progress to a different business model in which the individual application is no longer in the foreground. Instead the document (the plan) becomes the point of focus once again. Remember, this used to be the case up until the introduction of the computer and software into the planning process.

Various groups may start to develop specialized components in place of what used to be one behemoth CAD application. Programs of various kinds, such as 2D drawing, 3D modeling, massing analysis and rendering software would share that common standard format. Just the same specialized applications such a bidding and tendering software or thermal simulation software would be able to use the same set of data. Redundancies and errors would be minimized. It remains to be seen how quickly the industry adopts this new standard. Similar to the adoption of open standards, such as the set of Internet-protocols or the universal database query language SQL, this is primarily a political issue requiring large governmental players to force the open standard into the marketplace.

Customers would increasingly want to purchase only those functions in their CAD applications, with which they work, and therefore require. In short, they would start to demand modularity in their CAD applications. This approach would also introduce for the first time, a market competition at the "function level." Currently, entire CAD empires compete with one another. Once customers have bowed their heads in the direction of one of these fiefdoms, their choice of functionality is usually dramatically limited. The customer has one set of tools with which to work and little if any alternative.

A new modular approach and implementation would allow the choice of individual tools required for specific design tasks. Users will be able to scale software solutions to their own desire and financial abilities.

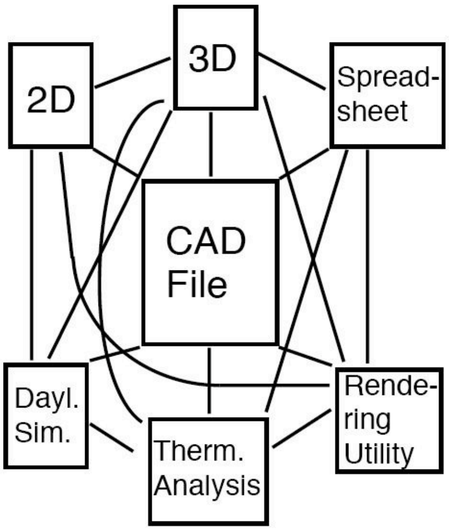

Figure 3: Free exchange of information in a modular, data-centric model

This approach would foster the development of dramatically more modular software. The traditional CAD-application, which we know today, may change to a mere *container* for nearly infinite plug-ins that the user purchases independently. Strategies like this are entirely possible, and are even executed by the industry. Software applications like Adobe PhotoShop® are almost unthinkable without the myriad of plug-in filters that come along with the application and are offered by independent vendors. Even open source applications such as the Mozilla Firefox project permit the addition of specific functions through the add-on of various plug-ins. AutoCAD and most of the other CAD applications usually offer programming interfaces, by which the software may be expanded upon from the outside.

This container application will contain all the basic intelligence in order to provide a platform for the internal data interchange between the various components, such as tools and (design) objects. These CAD-container applications may be distributed for free, similar to the PDF format where the Acrobat Reader application may be freely downloaded by anyone. So, while the toolbox itself may not cost anything, users would start to pay for the tools that they plug into the tool chest. On the other hand, different vendors, may sell this tool chest with a starter set of tools for a moderate price and offer additional functionality as add-ons.

All applications, however, would follow the same internal program interface guidelines (though not developed yet, this may be yet another standard outgrow of the IFC initiative) in order to stay compatible and make the tools completely interchangeable between different applications. Just as one used a traditional compass here, there, or somewhere else, one may use a little compass application plug-in, in whatever CAD-application. For the first time this would give software users *choice*, one of the mantras of a free market society.

Standardization would allow for all the tools within such an application to be little applications of their own, which could be selected individually and easily plugged-in and plugged-out of the "mother ship" so to speak. This approach would cater to the designer's needs to custom-tailor the software and its capabilities determined by individual user requirements. The Internet provides an excellent medium for direct sale between the developers and customers. As the tools would usually be fairly small in size, they could easily be downloaded from the net.

Standardization on the software side is only a logical move after the standardization efforts on the hardware side. "Fast networks may assist realization of a vision in which the machine one works on only plays a role in local graphics processing. Otherwise the local machine may easily convert into a virtual super computer by accessing other processors via the network. The analogy to the power grid becomes evident now: nobody in industrialized nations, apart from special cases, runs his own power plant. We are all used to consuming different amounts of electricity via the power grid." (Schmitt, 1993).

This development could well bring about considerable expansion within the CAD industry including a shift away from monolithic and inflexible applications. It would allow small vendors to concentrate on specialized tools without the need to invent the entire CAD-application around it.

It would permit for more rapid introduction of new ideas and concepts into the industry as the pace of development is no longer controlled by a few software giants that may follow a different agenda than what the building industry and its participants would want to head towards.

2.2 EFFECTS ON EDUCATION

Most Schools of Architecture stand at the receiving end of this development today, unfortunately. Not being in the field of software development, they have to pretty much accept anything the industry throws their way. Development of open standards would benefit the academic sector as well. No longer would schools of architecture and engineering face the walk on a thin wire of more or less openly acting as sales agents for private software companies. CAD applications are difficult to learn and research shows that professionals frequently are reluctant to changing away from software applications that they learned in the university unless the new applications offer dramatic improvements and larger capabilities. Given the complexity and steep learning curve this complacency is an all to understandable human behavior. The reason why CAD-software vendors usually offer substantial academic discounts is not entirely altruistic as the software companies try to bind their future customers early on. This behavior is not unlike the illegal drug industry. Unfortunately it stabilizes the status quo as opposed to fostering innovation. Once the free flow of information is established through an open file exchange standard, the software wars in schools of architecture should be an issue of the past, replaced by a re-emergence of focus on content rather than technology, semantics rather than syntax.

Schools of architecture and engineering may shift their focus away from individual CAD-applications towards a more holistic approach of digital media and their underlying philosophies. This would dramatically expand the shelf life of the knowledge they transmit. While individual software applications are currently updated at 12-18month intervals and their vendors would like users to believe that they will fall hopelessly behind the technology curve if they do not continuously upgrade to the newest versions, their underlying philosophies have not changed dramatically in the last two decades. CAD applications still use geometric entities to describe objects and they usually use layers to organize these drawings.

The only major paradigm change the industry faces at the moment in the field of building CAD is the shift away from simply drawing 2D or 3D shapes towards designing real building entities using real construction objects inside the computer and trying to simulate the building as closely as possible prior to construction.

The shift away from the monolithic software model would liberate the users as the change of software becomes similar to the changing of car models. While each new generation of a specific automobile may embody improvements over the previous, improvements in safety, comfort, power, fuel economy, etc. the basic concept is not put in question. If one can drive the current model one should be very able to drive the future model just the same. This extension of the shelf life of knowledge is of enormous importance to students, the profession, as well as, to the academic institutions. In such a scenario the body of knowledge of what is an architect should become dominant once again over what software the individual prefers. This would allow Schools of Architecture to turn away from the race with the software vendors and be permitted to focus on the topic of properly designing buildings once again.

3. Conclusion

There are many trends in the computer industry today. While certain software organizations appear to be firmly established or just solidifying their position in the entire software industry, independent standardization bodies such as the ISO, ANSI, DIN or the W3 Consortium, more than ever before, are trying to establish open and public standards. The development and distribution concept of the open standard Linux operating system currently poses the biggest threat towards established commercial software, and the dependency of the user to large software corporations. This is a good reason for optimism with respect to the opportunities for the development of open standard CAD applications.

Architects and engineers, for the sake of the survival of their profession are called upon to, once again, set the rules for the exchange of information in their domain.

This way, eventually and hopefully, the discussion over what CAD-application is in use or that one can operate becomes as important as the daily talk about the weather or the dispute over the most recent architectural fad: important but not earth-moving.

References

AMERICAN INSTITUTE OF ARCHITECTS (AIA) http://www.aia.org/

BAKER, ROBIN, 1993. Designing the Future, The Computer Transformation of Reality London: Thames & Hudson, Here: German edition, Duesseldorf: Econ Verlag, 1993, p.31.

BIRX, GLENN W.: http://www.aia.org/ aiarchitect/ thisweek05/ tw1209/ tw1209changeisnow.cfm

BUILDING SMART: http://www.iai-na.org/bsmart/

DAVIS , DIANNE: http://www.aia.org/tap_a_0903bim

http://en.wikipedia.org/wiki/CAD_data_exchange#Data_translation_formats

"General Introduction to STEP (ISO 10303)" http://www.nist.gov/sc4/www/stepdocs.htm

SC4 Standards http://www.tc184-sc4.org/About%5FTC184%2DSC4/About%5FSC4%5FStandards/

INTERNATIONAL ALLIANCE FOR INTEROPERABILITY: http://www.iai-na.org/

List of Certified Applications: http://www.buildingsmart.de/2/2_01_01.htm

SCHMITT, GERHARD: Architectura et Machina, Computer Aided Design und Virtuelle Architektur, Braunschweig/Wiesbaden, Vieweg Verlag, 1993, p. 225, translated by author

TARDIF, MICHAEL: http://www.aia.org/tap_a_0903cad