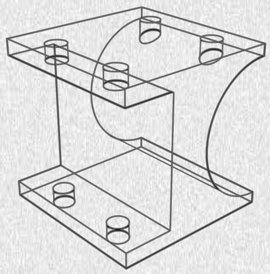
IronCAD



Not all Design Methodologies are Created Equal

Not all Design Methodologies are Created Equal

Executive Summary

When selecting your next design system, it is critical to understand the various design methodologies adopted by each of the CAD products available today. It is important to understand that not all design methodologies are created equal; each one has both strengths and weaknesses, making it vitally important that a company matches the right design methodology to the types of products they produce and the design culture of its engineering staff. Selecting the wrong design methodology can dramatically affect the overall design productivity of the company.

Parametric based systems strengths and weaknesses.

(Parametric Modeling, History-Modeling, and Feature-Orientated Modeling)

Strengths	Weaknesses
• Ability to capture design intent	• Rigid requirement of parent child relationships
• 3D capture of existing designs	• Upfront knowledge of "Design Intent" requirement
• Support for family of parts design	• Requirement of replaying full history to view changes
	• Increased model size decreases ability to innovate
	• Legacy model knowledge within company
	• In context editing

Non Parametric based systems strengths and weaknesses.

(Explicit Modeling, Direct Face Modeling, and Boolean Modeling)

Strengths	Weaknesses
• Guaranteed that what you edit is what you end up with	• Unable to maintain individual l feature identity
• Ability to edit geometry imported from other systems	• Unable to re-order feature history
• Faster regeneration times	• More difficult to capture design intent

IronCAD's innovative Freedom Architecture™ is the only one that combines the positive aspects of parametrics while providing alternate innovative ideas to minimize the negative side. IronCAD provides not just one but a set of modeling methodologies for the user to select from during the design process.

IronCAD's superior architecture has allowed IronCAD users to enjoy tremendous productivity gains over what each customer experienced with their previous 2D and 3D CAD system. The numbers talk for themselves.

- Productivity improvements of using IronCAD™ over AutoCAD™ (2D): **191%**
- Productivity improvements of switching to IronCAD™ over other 3D applications: **45%**

- Competitive 3D System % reduction in product development time where the previous system was AutoCAD™ (2D): **28%**
- IronCAD's % reduction in product development time where the previous system was AutoCAD (2D): **68%**

Not All Design Methodologies are Created Equal

Designing in 3D is now standard practice for companies around the world. Deciding on which 3D CAD system is often a confusing process especially when confronted with the marketing messages of today's leading technology vendors. Companies over the years have tried to sort out this confusion using a spectrum of decision-making methods starting at one end with large-scale multiproduct CAD evaluations: All the way down to the other end where companies just pick the most popular system, relying on their assumption that so many other people cannot be wrong. Regardless of the method used, when selecting your next design system it is vitally important to understand the various design methodologies adopted by each of the CAD vendors products. It is important to understand that not all design methodologies are created equal each one has both strengths and weaknesses making it vitally important that a company matches the right design methodology to the types of products they produce and the design culture of its engineering staff. Selecting the wrong design methodology can dramatically affect the overall design productivity of the company.

The good news is that over time the solid modeling methodologies have boiled down to two primary ways of creating solid models, parametric or non-parametric. The decision of which method to use must be made before moving to the next step of looking at the vendors that represent that particular design methodology. To find out which method is best for you please read on.

Parametric Modeling

Parametric Modeling is a way of defining geometry through a set of geometric dependant parameters/constraints, so that editing one shape will cause other shapes to change size or location. For example, a hole that is supposed to be in the center of a face could be created that way with an equation, so that no matter how the face is resized, the hole always relocates itself to the center.

Parametric modeling was first introduced back in the 1980’s. Since then we have seen Windows™ versions of that technology introduced by the mid 1990’s. Since then parametric modeling has become the dominant design methodology in use today. Basically the reason being that is was the only game in town.

The parametric design methodology:

Parametric design systems require the user to work within a fairly rigid modeling process. First the user has to create a fully constrained 2D sketch. This sketch is then converted into a 3D object using one of several modeling operators such extrude or spin. This first feature of the model is usually referred to as the base feature. The user then repeats this process to add additional features to the base feature with the requirement that each new feature has to be positioned and size constrained relative to one of the previous features creating what is called a parent child constraint relationship. This concept of parent child constraint relationships is not just used for part creation it is the foundation upon which all parametric systems operate so it affects all areas of the software such as assembly creation, construction and datum plane placement, dimensioning etc. The result of this process is a model where a change to any one of the parent features can cause any number of child features to also change, often unexpectedly.

The positive side of parametric modeling

Ability to capture design intent – Since parametric systems require that one feature is dependent on another they are ideal for capturing what has come to be known as design intent. Design intent can be defined as the ability to capture form, fi t and functional requirements of a design during the modeling process: this is usually accomplished using various types of constraints.

3D capture of existing designs - Parametric systems are very effective at converting existing well know designs into 3D models: A design where the change parameters are well known and are not prone to change.

Support for family of parts design - Parametric systems are a perfect fi t for a family of parts design where the geometry just varies in size, position, and visibility. Parametric systems are not as strong in cases where input parameters lead to the creation of new additional geometry.

The negative side of parametric modeling

Rigid requirement of parent child relationships – Parametric systems require constraint relationships at every stage of the model building process forcing the user to add more constraints than what would normally be required for design intent, setting the stage for possible design problems down the road. There is power in capturing a certain amount of design intent (IE: a cylinder that has to stay a constant distance from a particular edge for example) but when you are required to add multiple, even hundreds of these interdependencies it grows difficult for a user to anticipate the accumulative effect that each relationship will cause. The negative impact of these kinds of parent child relationships can be unanticipated model changes or a model that fails to regenerate due to some form of constraint conflict. In either case a significant amount of time is usually required to resolve the problem resulting in a lower than expected design productivity.

Upfront knowledge of "Design Intent" requirement – Since parent child relationships are fundamental to all parametric modeling systems, it is vitally important that the user have a clear understanding of how a model is to function, commonly known as its design intent.

Without this prior knowledge (design intent) it is very easy for users of parametric systems to create a model that restricts the creative process and in some cases results in the model having to be recreated entirely, in order to accomplish a single desired change. Taking our example of a hole, which due to the parametric methodology was positioned a certain fixed distance from a particular edge using a dimension. What would happen if that edge moved? Well the answer would be the hole moved as well to maintain the constraint, but is that what the user wanted? Was the position of that hole important for some other reason? Imagine trying to know ahead of time how a model needs to change during the creative process; it is next to impossible. Even if you did know how it was to change, it would still be very hard to build a model where the accumulated parent child relationships worked as needed. A user would find that as the model complexity increases, the ability to innovate decreases.

Requirement of replaying full history to view changes – Since a parametric system creates a sequential relationship between each feature, for a user to view the results of any change the entire history tree has to be replayed. This is not much of a problem with smaller models but when the model becomes large this requirement can become very time consuming and frustrating. This frustration can be compounded when a recent change causes a failure somewhere in the history as it is being replayed. Fixing these failures in one –area often causes a failure somewhere else, often resulting in the user chasing a regeneration error around his model and losing valuable design time.

Increased model size decreases ability to innovate – Based on some of the previous negative side effects of parametric systems it is easy to see that the seriousness of these issues increases, as does the model/assembly size. As the model complexity increases the ability to innovate decreases. Modeling productivity is reduced as more and more time has to be spent working around previously created parent child relationships that are now interfering. You would like to remove a certain feature but you are prevented from doing so because some other feature references that feature. You would like to relocate a part by a certain amount but when you do so, all other features, parts and/or assemblies that are referenced to that part also move. More and more time has to be taken to make sure that any change that is to be made does not cause an unanticipated change somewhere else.

Legacy model knowledge within company – When a user creates a parametric model they have to fully understand the parametric relationships they have built into the model, so they can effectively perform any required changes. If they don't have this in depth knowledge it is almost certain that they will fall prey to one or more of the listed negative issues. This causes some immediate problems when a team of design engineers is working together on a modeling project. How does engineer B know what design intent engineer A built into his model when he has to include it in an assembly? What happens when engineer B has to edit a model that was created by engineer A, who is no longer on the project or has left the company? What happens when engineer C has to make a manufacturing change to an assembly that was built by engineer A and engineer B? Parametric systems make it difficult for interdisciplinary teams to work together when the parent child relationship knowledge is with the creator of the model but not other team members or downstream consumers of the design. Alleviating this problem requires the use of very strict design guidelines, which cause a time burden on the designer and a limiting factor to design creativity.

In context editing – Traditional parametric CAD systems create what is called a feature tree history, which reflects the order in which the features were combined to create the model. When you want to edit a specific feature in this parametric tree, all the features that occur after that point in the history are blanked out. All the user sees is the geometry up to the point of the feature to be edited. This approach creates a non-productive situation for the user because often the user would like to size or position the edit feature relative to one that is no longer visible.

Non-parametric modeling

For the case of this document, non-parametric modeling is all other modeling methodologies that do not require parent child constraint relationships. This includes boolean modelers where the model is created by adding or subtracting a set of analytic primitives in order to obtain a desired form; and direct b-rep manipulation modelers (explicit or direct face modelers) that create the desired model by adding and subtracting new face features or modifying existing faces that belong to a single part body. Boolean and direct b-rep manipulation modelers existed prior to the introduction of parametric modelers back in the late 1980's. Some boolean modelers that are still in use today. In recent times (2008- 2009) the Direct Modeling approach has regained some momentum for the fact that the industry has realized that parametric modelers are simply too much work to be productive. These "new" direct or explicit modelers resurged with a few "new" tricks and a host of creative trademarked names such as "Synchronous Technology", "Fusion Technology" and more.

The non-parametric design methodology:

The main distinction of non-parametric design systems is that they do not maintain a feature history or require parent child constraint relationships between each feature. For example if you place a cylindrical hole through a block in a Boolean modeler, the only way to move that block is to first fill up the original hole and then create your new one. In the case of a b-rep modeler you may have added a cylindrical hole initially but on a subsequent edit it does not recognize it as a hole but as a face belonging to a single brep model: You can move it as an independent face but if that hole intersects with other features of the part each remaining segment of that cylinder becomes a separate b-rep face, with no understanding that all the faces that originally belonged to the cylinder are still associated. Since non-parametric modelers do not have a history tree there is no concept of feature re-ordering which is a common modeling practice in parametric modelers.

The positive side of non-parametric modeling

Guaranteed that what you edit is what you end up with – Since there are no parent child relationships to worry about, edits only affect the face(s) being edited. It is a true WYSIWYG editing paradigm.

Ability to edit geometry imported from other systems – Since most geometry passed between systems goes through a neutral format, it loses all feature history and becomes what is called a single b-rep shape. Since this is the base method of editing used by direct b-rep modelers, they can handle this type of geometry very effectively. Parametric modelers on the other hand have to rely on feature recognition, which often fails: Thus they are unable to edit at the face level.

Faster regeneration times – Since non-parametric modelers do not have to replay the entire feature history in order to view the latest model change they tend to be faster when editing larger models.

The negative side of non-parametric modeling

Unable to maintain individual feature identity– Maintaining feature identity allows faster editing to be performed especially in cases where several features may intersect. Take for instance a cylinder that may intersect several slots cut into a block: In a parametric system you would move the cylinder to a new location (barring parametric constraints). All the faces belonging to the old hole in the parametric system would disappear and be replaced by the new hole in its new location. In a direct b-rep modeler the user would be responsible for selecting all the faces that belonged to the original cylinder and performing a move on the selected set. The process of selecting all of the correct faces especially where internal faces that are not visible to the naked eye are involved, takes significantly longer than just moving the cylinder. With a Boolean modeler approach you would have to add a solid cylinder to fill in the previous hole and then add a new cylinder in the new location; thus almost doubling the amount of time that is required to perform a simple feature move in a parametric modeler (barring parametric constraints).

Unable to re-order feature history– A parametric system's ability to change the order in which features are applied, gives the user a way to quickly change the final geometry outcome. Take for instance a boss that was added to a previously shelled body. The boss would be solid because it came after the shell. To make it so the boss was shelled, the user simply has to move the boss before the shell in the history tree. This simple operation cannot be performed in a non-parametric modeler. The user would have to apply a more complex alternate method or undo the last set of modeling steps, then add the boss and re-apply the shell.

More difficult to capture design intent – Many times a designer would like to capture design intent while building the model. What comes naturally in a parametric system has to be captured in a secondary step of adding constraints. In many cases this might not even be possible because a feature intersection has broken the faces of the original feature into several separate faces that cannot be constrained to move together.

Which Design Methodology is Best for Me?

After reviewing the positive and negative sides of both design methodologies the time has come to decide which method is best for you and your company? Like most people you are probably asking the question, "Why can't I have it all?" There has to be a solution out there that gives me the best of both worlds. The good news is that there IS a solution out there that can give the benefits of both and its name is, IRONCAD™ (Industrial Revolution ON Computer Aided Design).

Historical Background of IronCAD

A company called Visionary Design Systems out of Santa Clara, CA first introduced IronCAD into the market in 1998. IronCAD was built upon a technology that they acquired when they merged with a company called 3D/EYE in 1997. IronCAD was hailed as the first new technology to hit the market since the late 1980's. It won numerous awards for its technological innovations and was the fastest growing CAD product at the time. Visionary Design Systems later changed its name to Alventive and chose to change the company direction towards web based design collaboration tools in late 1999. In March 2001 several executives and other employees split away from Alventive and created IronCAD, LLC. The goal of the company was to once again restore IronCAD as a leading 3D design tool and to help people understand it's potential. Alventive has since closed its doors in 2002.

IronCAD's design methodology

Since IronCAD was developed in the mid to late 90's, both the positive and negative aspects of parametric systems were known. The goal of IronCAD's development team was to create an architecture that would enhance the positive aspects of parametric systems while minimizing the negative. To achieve this goal, the development of a very flexible architecture that could operate from several different design methodologies was required. What this did was give the user the power to select what design methodology to use to best accomplish the task. Gone was the rigid requirement to always do things the parametric way. The freedom of choice was a key design objective of IronCAD's new design methodology.

The IRONCAD™ key elements of innovative design architecture are:

- The freedom to model a part without knowing ahead of time what future changes might be required.
- The freedom to choose whether or not to start from a 2D sketch or from a pre -built feature, part, or assembly.
- The freedom to choose to apply constraints to a 2D sketch or not.
- The freedom to choose to create multiple parts or assemblies within the same file or not.
- The freedom to choose to create a feature history tree or not.
- The freedom to choose to edit a part using features or through direct face manipulation.
- The freedom to choose to position features, parts and assemblies using dynamic soft or permanent constraints or without constraints entirely.
- The freedom to choose to selectively collapse portions of the feature history of a part when it gets in the way, or not.

- The freedom to choose to get the true history model, or one that provides an alternate result.
- The freedom to edit a feature of a part while relative to all the other features of the part even if those features occur after the selected feature in the history tree.
- The freedom to add or remove a constraint without the fear of causing a regeneration error in the model.
- The freedom to have facet parts (STL, VRML, 3DS) as well as b-rep solid parts in the same assembly.
- The freedom to freely edit a part or assembly from a colleague without the worry that some unknown change will occur.
- The freedom to choose to edit a feature through the use of dynamic handles rather than having to constantly edit the 2D sketch plane or not.
- The freedom to precisely position and size features, parts and assemblies without having to type in a constant value or use constraints.
- The freedom to chose which kernel to use for a particular project (ACIS®, Parasolid®). By Default, IronCAD uses both simultaneously.
- The freedom to both create and modify features, parts and assemblies relative to each other without the restrictions imposed by parametric systems.

Some Key Unique Features only IRONCAD™ provides:

Multiple Modeling Methodologies - IronCAD provides the best of all worlds. It provides "Structured Mode" which is its History constrained mode; and "Innovative Mode " which is Non-History constrained mode.

> *Structured mode* acts much like all other parametric modelers using a history tree and constraint system, however of course IronCAD has added tools to make it even easier than its closest parametric competitor.
>
> *Innovative Mode* is schizophrenic, it basically allows you to work in any way you want, even as parametric system if desired. Innovative mode allows you to combine all modalities such has feature tree orientated, parametric, explicit (direct face) , boolean, and IronCAD's unique ability to hybrid all of them together.

Dual Kernel Technology - IronCAD developers mastered the ability to have the two leading kernels (ACIS® and PARASOLID®) that the majority of all other CAD systems use into a simultaneously synchronized solution. IronCAD developed this close to ten years ago and is still the only application to offer this. Basically no one can figure out how they did it.
The two modeling kernels like CAD software have their strengths and weaknesses. For instance ACIS® is extremely good at modeling parts that contain a high degree of tangents, whereas PARASOLID® would fail. PARASOLID® is very good a blending whereas ACIS® would fail. The

majority of 3D CAD applications on the market together use one or the other. So in effect they have a high model failure rate because they are limited to one kernel. With IronCAD you have them both working for you full time, thus virtually eliminating model failure due to kernel limitations.

TriBall™ - The TriBall is the holy grail of IronCAD and basically 3D CAD design in general. IronCAD holds the patents on this feature and hence many competitors have tried to duplicate it in their own programs, but falling short every time. The TriBall in a nutshell eliminates the need to rely on cumbersome tools, datum planes and otherwise to manipulate parts and assemblies in your design in very complex 3D spatial situations.

SmartSnap with Handle Technology - Once again IronCAD developed the handle technology over ten years ago, many try to mimic but cannot come close to the abilities found in IronCAD. Using handles on geometry along with SmartSnap technology provides a powerful modeling function to quickly create accurate and precise designs without the need for cumbersome dialog entries.

Completed 2D/3D Associatively - With the release of IronCAD 2009 XG, IronCAD has integrated a 2D standalone drafting application (CAXA Draft) that functions very similar to the leading 2D CAD applications on the market today. IronCAD has made this complete associative to the 3D data which makes it the only application on the market today to do so. Competitors have very lightweight comparisons to this approach.

Multitude of On-Demand Technologies - IronCAD flexibility to choose what and where to assign a task or function works hand and hand with its on-demand technologies. On-demand refers to the ability to apply a tool, constraint, feature, etc as needed instantly. Some of these tools include; Auto-Feature™ recognition, positioning constraints, 3D dimensional constraints, history-toggling, rendering, and more.

Selecting the Best Tool for the Job:

The key to maximizing a company's 3D design productivity is to insure that a product is selected whose underlying design methodology matches the types of products that the company makes. It is understood that the design methodology of the software product is not the only criteria that has to be taken into account in the decision process. The scope of this paper is not to educate a user in the total decision process, but to educate the user with regards to the possible affects to their organization depending on the chosen design methodology.

A key point to remember is that every product has strengths and weaknesses: The trick is to find the product whose key strengths best match the type of work that you perform most often.

The table shown on the first page of this document can be used as part of that decision process. All the user has to do is put the percentage of time they spend in each activity. By looking across at the design methodology columns the user will be able to select the 3D design methodology whose key strengths match where the highest percentage of their time is spent. This method will ensure the highest possible design productivity from the 3D design tool as part of the overall equation.

Matching Design Activity to Design Methodology

Design Activity	% Time Spent	Parametric	Non-Parametric	IRONCAD™
Parametric Modeling		●	○	●
Family of Part Design		●	○	●
Moving known 2D designs to 2D		●	○	●
Modifying existing designs		○	●	●
New Assembly Design		○	○	●
New Part Design		○	○	●
Unanticipated change handling		○	●	●
In-context part/assembly editing		○	●	●

○ Possible but not key strength ● Key Strength for methodology

In Summary

The underlying design methodology of the 3D design tool can have a dramatic impact on the design productivity of a company. Selecting the right design methodology for the type of work performed can lead to exceptional productivity gains. Each design methodology has inherent strengths and weaknesses. Mapping design methodologies strengths to the type of work performed is an important step in the process when deciding upon a new CAD system. Systems that utilize the parametric design methodology are very good at detailing an existing design where all the change parameters are clearly understood ahead of time and are not prone to change. They especially are a good fit for a family of parts design where the parameters entered as part of the parent child constraint relationship process can be presented as edit points for the creation of various part configurations, which are common in this type of modeling. Due to their parametric nature they are very good at capturing design intent, maybe even too good since every action is constrained rather than just the design intent desired by the user. They are not as good in areas of the design process subject to radical changes such as in the early conceptualization phase or in late manufacturing change situations. Parametric modelers struggle when working in situations where a lot of unanticipated changes may occur. Since a lot of care has to be taken when creating a parametric model the learning curve of these systems is generally measured in months. It takes much trial and error before enough experience is accumulated to avoid the more common types of model regeneration problems.

IronCAD's new Freedom Architecture™ is the only one that combines the positive aspects of parametrics while providing alternate innovative ideas to minimize the negative side. IronCAD provides not just one but a set of modeling methodologies for the user to select from during the modeling process. In the early phase of the design, IronCAD can work very conceptually using its handle -based technology. As the design matures the user can migrate to more precise techniques that might require the use of constraints to capture certain elements of the design intent. The freedom to alternate between these various approaches is what makes IronCAD so productive.

As the 3D CAD market matures you will find that most 3D systems available today can handle 95% to 98% of what is thrown at them. The decision that is left is which system can do my parts most productively? Most people look at which systems are biggest in the market place but that measure might not always lead to the best choice.

Choosing which of the available design methodologies is best for your company is an often overlooked criterion. None of the parametric based system providers will tell you what the negative aspects are of their approach; you have to find out by yourself. After reading this document you will be exposed to the issues and the best available alternative, IronCAD.

The productivity numbers of IronCAD are impressive when compared to the best systems out there today. The only way to fully appreciate the IronCAD Difference is to take IronCAD for a test drive or contact Magnacad, LLC to show you the IronCAD Difference one on one. Its time you found out for yourself why...

IRONCAD™ provides "Limitless Design Dexterity"

magnacad

Magnacad is a major supplier of the most innovative and mission critical software for design professionals and enterprises nationwide. We have been involved in the industry over thirty years and have seen a need to provide services and solutions that are a dramatic change to the way organizations maintain their design and engineering infrastructure.

www.magnacad.com

Magnacad, LLC
169 Commack Road, #160
Commack, NY 11725

(631) 974.0677