

# REINVENTING 2D CAD:

## ENABLING SWIFT AND SMART DESIGN



*Designing with 2D geometry is now seen as a legitimate way to develop new concepts and ideas. Now there is a new alternative to Drafting Tools and Feature-based Sketchers that offers the ability for swift and smart design with 2D geometry: Direct Sketching.*

***Published by:***



# REINVENTING 2D CAD: DRIVING SWIFT AND SMART 2D DESIGN

## Designing with 2D geometry is legit.

Just a mere five years ago, publically saying such a statement would get you jeered and mocked. The prevailing attitude was that designing with 3D models was the highest level of engineering maturity. Mucking around with 2D geometry was seen as a laggard practice, only used by the stragglers of the industry. Since feature-based modeling first emerged some two decades ago, there was an incessant push to move from working with 2D geometry to building 3D models.

Now, it's fairly obvious that producing engineering drawings with 2D geometry isn't efficient compared to laying down views of 3D models. Few would argue against that. But thankfully, more progressive thinking in the industry has surfaced in the past five years. Engineering organizations and software providers alike have grasped the critical nuance between documenting and designing. And in that light, designing with 2D geometry is a legitimate way to iterate and explore ideas in concept design and early detailed design.

Fast-forward from then to now, and design practices have changed. In early design, engineering works with 2D geometry, leveraging it to make critical decisions. That work with 2D geometry then becomes the basis of 3D models in detailed design. There is a good continuous transition from one to the other. Overall, it's astounding that attitudes have changed so noticeably in such a short time.

When it comes to the technologies that enable design work with 2D geometry, many of today's CAD software applications provide both Drafting Tools and Feature-based Sketchers. Unfortunately, each of them poses unique problems when faced with the context of such

early design efforts. Designers and engineers often use released engineering drawings or cross sections from finalized 3D models, each of which contain hundreds if not thousands of lines, curves, arcs and splines.

Drafting Tools allow them to trim and extend those 2D geometric entities on a one-by-one basis. It does not, nevertheless, power intelligent change. On the other hand, Feature-based Sketchers enable smart modifications. But they inhibit the iteration and exploration of new ideas with their need to full define the location of every 2D geometric entity with constraints and dimensions.

Fortuitously, an alternative for designing with 2D geometry has arrived. Direct Sketching, which applies Direct Modeling concepts to 2D geometry, enable intelligent change without inhibiting design iteration and exploration.

Therein lies the purpose of this eBook. It contains more background on the keys to successful early design, further details on the failings of Drafting Tools and Feature-based Sketchers for designing with 2D geometry and the critical capabilities of Direct Sketching.

So read on. Designing with 2D geometry is legit. Its time designers and engineers get the right tools for it.



# REINVENTING 2D CAD: DRIVING SWIFT AND SMART 2D DESIGN

## The Sensibility of Designing with 2D Geometry

Where exactly does it make sense to use 2D geometry in early design? There are a couple scenarios where it is simply more pragmatic instead of building 3D models. Let's look at each in turn.

### Quick and Simple Evolutionary Design

Let's face it. When it comes to modifying an existing design, sometimes there is no 3D model to change. More often than not, the only existing digital asset is the engineering drawing that is composed of 2D geometry.

There certainly are cases where building a new 3D model for the design is warranted. However, if the required design modifications are simple, then updating the 2D geometry on the engineering drawing to capture the change represents a quick and easy solution. This is a pragmatic option to keep the project moving forward. And ultimately, that's the purpose.

### Exploration of Ideas in Concept Design

Another good fit for designing with 2D geometry are a few scenarios in concept design. This might be generating different ideas for a product prior to kicking off the development project. It might be exploring some alternatives for a component before committing one to detailed design.

Either way, working with 2D geometry offers a quick way to build out some minimal representation of the design that the designer or engineer can use for their comparative assessments. The 2D geometry used for these activities can later be leveraged to build the 3D model in detailed design.



# REINVENTING 2D CAD: DRIVING SWIFT AND SMART 2D DESIGN

## Using Drafting Tools to Design with 2D Geometry

One option to design with 2D geometry is to go old school by using Drafting Tools. These are the same tools that have been used on engineering drawings for decades now. They enable designers and engineers to trim, extend, copy, delete and perform other operations to 2D geometry like lines, curves, arcs and splines.

### Modification Capabilities that Don't Scale

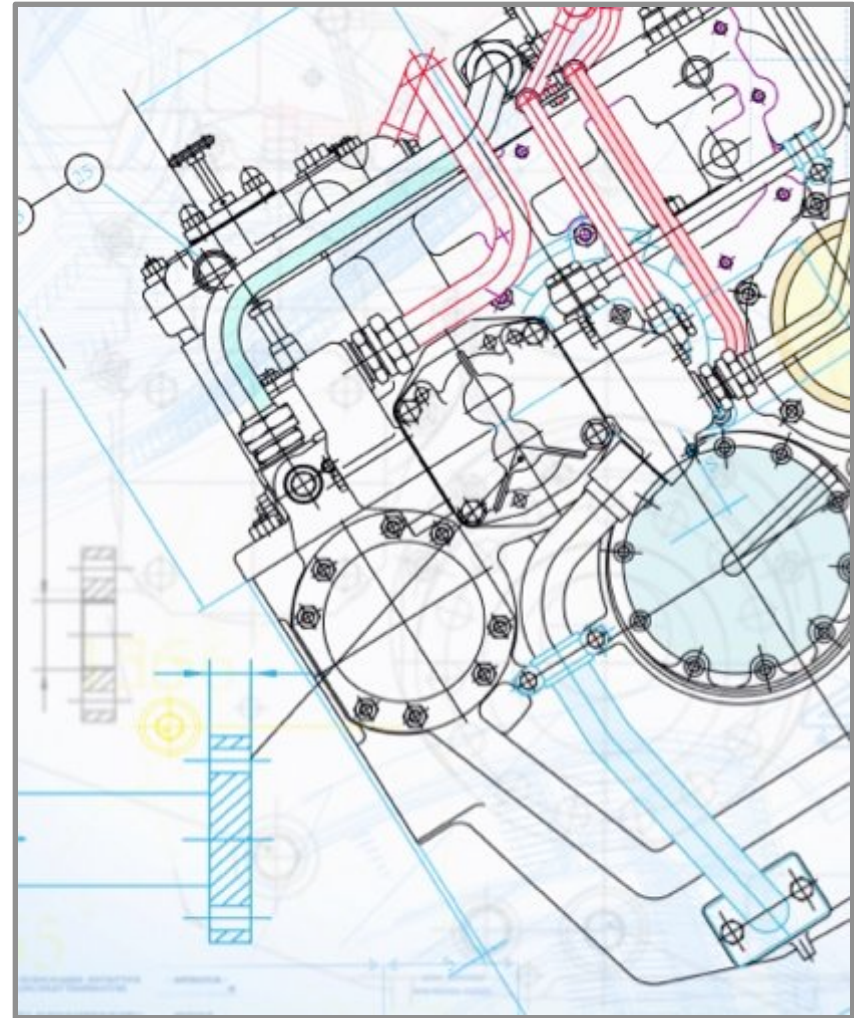
What is the issue with this approach? The first problem lies in the amount of time it takes to create or modify the 2D geometry. Today's designs have a large number of entities, numbering in the hundreds or thousands. With Drafting Tools, the lines, curves, arcs and splines that make up the design have to be modified on a one-by-one basis, even if they are connected as a chain and should actually be considered a single continuous entity. In all, the Drafting Tools inability to drive broader change undermines its ability to be an effective support design with 2D geometry. It simply takes too much time.

### Capturing and Enforcing Design Intent

Another issue encountered when using Drafting Tools to design with 2D geometry is the inability to capture and enforce design intent. Drafting Tools offer no ability to apply user-defined rules such as parallelism and concentricity to the lines, curves, arcs and splines of a design.

With every modification, designers and engineers have to remember what design intent to consistently enforce, hoping they don't forget one along the way.

All in all, Drafting Tools don't enable fast explorative design with 2D geometry. They certainly provide productive capabilities for editing engineering drawings. But they simply aren't a good fit for the modern needs of design.





# REINVENTING 2D CAD: DRIVING SWIFT AND SMART 2D DESIGN

## Using Feature-based Sketching to Design with 2D Geometry

An alternative to Drafting Tools for designing with 2D geometry are Feature-based Sketching, which create sections that are used to extrude, revolve or sweep to create or remove solid geometry in 3D models. Before we determine if sketchers are or are not a good fit for 2D design, it's important to understand how they are different from Drafting Tools.

### Feature-based Sketching Require Defined Sections

---

In Drafting Tools, the location of each 2D geometric entity is precisely known as it is created. When Feature-based Sketching are used, alternatively, the location of each 2D geometric entity is abstracted. This is done in order to provide parametric control of modifications to the section. As such, principles long used to detail engineering drawings are applied. The location of each 2D geometric entity must be fully defined through some combination of dimensions and constraints.

Designers and engineers can certainly manually create specific combinations of dimensions and constraints for sections. Though it is far more common that Feature-based Sketching auto-dimensioning capabilities are used to do this automatically for the section. In fact, this automated capability isn't a one-time action. Modern sketchers often enforce, reapply and adjust those definitions schemes to sections in real-time, even during modifications.

These capabilities are powerful for sections that are used in features to build 3D models. But are they right for developing layouts with 2D geometry?

## As 2D Geometry Totals Rise, Performance Degrades

---

As mentioned early in this eBook, designers and engineers often use released engineering drawings or cross sections from finalized 3D models, each of which contain hundreds if not thousands of lines, curves, arcs and splines.

Feature-based Sketching enforce, reapply and adjust combinations of dimensions and constraints for sections in real-time. For sections with tens of 2D geometric entities, there are few problems. But once the numbers rise into hundreds and thousands, performance degrades rapidly. They can even become completely unresponsive.

### Hindering Design Exploration and Iteration

---

The requirements for Feature-based Sketching to full define sections don't just cause performance issues, they also hinder attempts to explore and iterate on designs. It is not unusual, given a certain combination of constraints and dimensions, that specific modifications of 2D geometry are impossible. In such cases, the defining scheme of the section must be changed before the modification can be made. Interestingly enough, this scenario is similar to building up a 3D model with numerous individual features, which then make it difficult to modify or even cause failures.

Fundamentally, the purpose of any design effort is to explore different options and alternatives. By exploring more alternatives, designers and engineers come across better designs.

CAD software applications shouldn't make those efforts harder. They should make them easier, especially when designing with 2D geometry.

# REINVENTING 2D CAD: DRIVING SWIFT AND SMART 2D DESIGN

## Using Direct Sketching to Design with 2D Geometry

To date, designers and engineers have had to compromise when picking between Drafting Tools and Feature-based Sketchers to design with 2D geometry. Now, they have a new option in Direct Sketching, which enables smart change, offers no predefined constraints and perform well even with large numbers of lines, curves, arcs and splines. Here are the critical capabilities of Direct Sketching.

### Explicitly Locating 2D Geometry

---

Earlier in this eBook, understanding how Drafting Tools and Feature-based Sketchers locate 2D geometry was critical to comprehending how those tools can then manipulate them. So let's start there with Direct Sketching.

Unlike Feature-based Sketching, Direct Sketching does not abstract and then locate 2D geometric entities with combinations of dimensions and constraints. Instead, they use a method more similar to Drafting Tools, where locations are already explicitly known in space.

By avoiding the sketching tool approach of enforcing, reapplying and adjusting those definitions schemes to sections in real-time, the performance of Direct Sketching does not degrade.

### Temporary Instead of Persisted Assumptions

---

When designing with 2D geometry, an important objective is to drive smart change instead of modifying individual lines, curves, arcs and splines. Without some

combination of dimensions and constraints that locates every 2D geometric entity, how does Direct Sketching drive smart change?

The answer is that Direct Sketching does in fact make assumptions, much like dimensions and constraints used by Feature-based Sketching, about the relationships between 2D geometric entities. These assumptions, however, are only *temporary* instead of *persisted*.

More specifically, during the modification of 2D geometry, Direct Sketching applies and enforces assumptions to enable smart change. Once the modification is complete, however, those assumptions are removed. Initiating a different modification creates a new set of assumptions that enable that change.

Therein lies one critical difference between these tools. Feature-based Sketching remembers the combination of dimensions and constraints for a *section*, constantly trying to enforce, reapply and adjust in real-time. Direct Sketching only creates and uses assumptions during the change, and not otherwise.

**REINVENTING 2D CAD: DRIVING SWIFT AND SMART 2D DESIGN**

## Local Instead of Global Assumptions

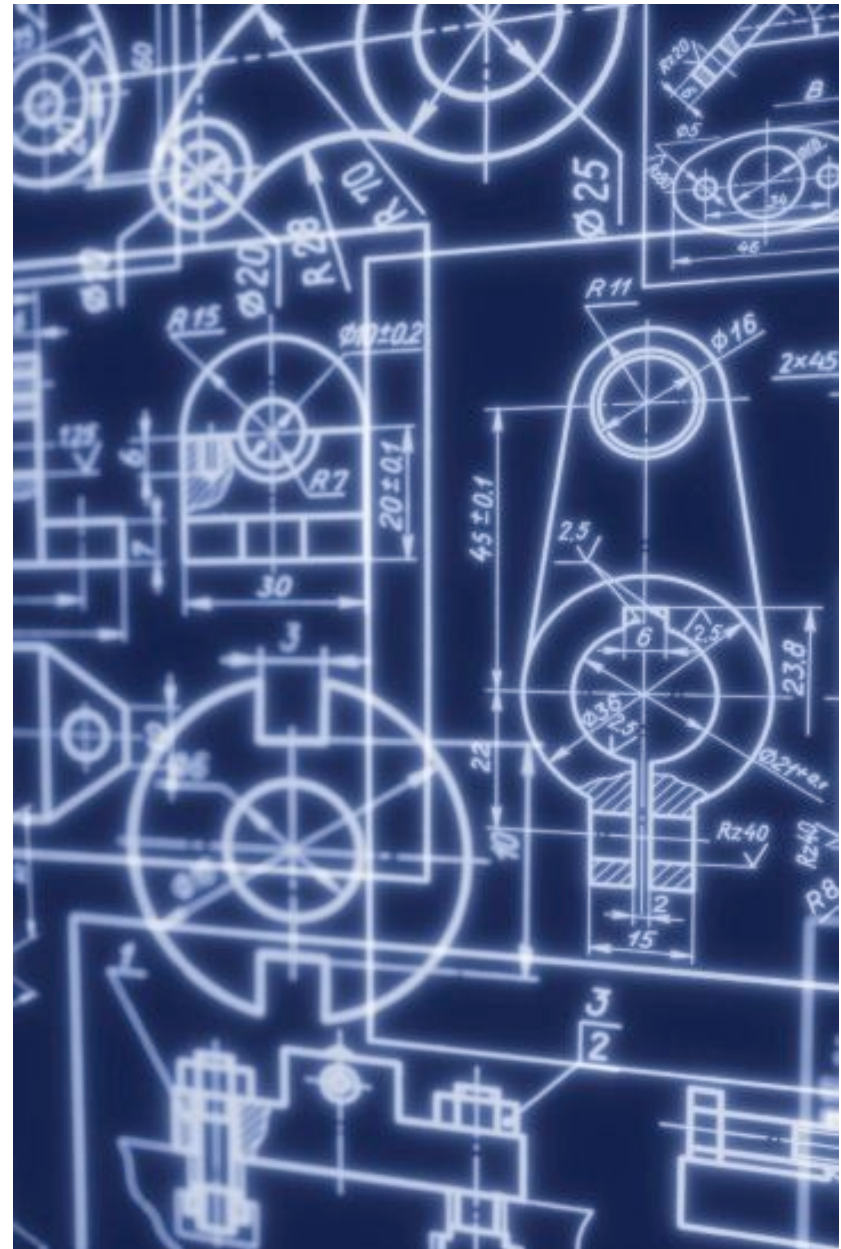
Applied the same set of 2D geometry, temporary as opposed to persisted assumptions offers a negligible advantage in terms of performance. In fact, the approach of creating and using new assumptions for every change would further impair performance compared to the one time definition of assumptions by Feature-based Sketching. The difference, however, lies in the scope of the assumptions.

Instead of developing *global* combinations of dimensions and constraints as Feature-based Sketchers do, Direct Sketching instead only develops assumptions *local* to the 2D geometry selected for modification. That, in turn, means that Direct Sketching analyzes far fewer 2D geometric entities for assumptions than Feature-based Sketchers analyze for dimensions and constraints. That offers a performance advantage.

## Defining Global Design Intent

Of course, temporary and local assumptions offer advantages in the form of swift and smart change to 2D geometry. However, it is also important to be able to apply and enforce design intent.

With Direct Sketching, designers and engineers can manually create dimensions and constraints. Even if they apply to 2D geometry that is not local to each other, Direct Sketching enforces them. This offers a means of embedding design intent in 2D geometry alongside the advantages already described for Direct Sketching.





# REINVENTING 2D CAD: DRIVING SWIFT AND SMART 2D DESIGN

## The Value of Swift and Smart 2D Design

So far, this eBook has covered how designing with 2D geometry is sensible for concept design and early detailed design. It has detailed shortcomings of both Drafting Tools and sketcher tools. It has introduced Direct Sketching, which offer advantages in terms of intelligent change without performance degradation. Of course, it wouldn't be complete without understanding the value, both to the company and the individual, of designing swifter and smarter with 2D geometry.

## The Value of Better Designs for the Company

From an organizational perspective, designing swifter and smarter with 2D geometry means designers and engineers can truly explore the design space of their products and components. Assessing more options means that the organization is much more likely to find a better design than what was original considered.

What does a better design mean for a company? That's a difficult question to answer specifically. For some, it means that the product outperforms all others in the market. For others, it means lowering the costs within the product, raising company revenues. For yet others, it can be difference between complete success and utter failure. While answer that question specifically is practically impossible, it's fairly safe to say that the outcomes are almost all positive and meaningful in one way or another.

## The Value of Faster Design for the Individual

While there are undoubtedly benefits of designing swifter and smarter with 2D geometry for the company, let's not forget about the designers and engineers. Today, they often have to meet tight deadlines in project schedules by working late and on the weekends. For designers and engineers, exploring designs and proving them out quickly can cut into that ugly sacrifice for work.





# REINVENTING 2D CAD: DRIVING SWIFT AND SMART 2D DESIGN

## ***Summary and Conclusion***

In the past five years, there has been a pragmatic admission in the industry that designing with 2D geometry is a legitimate practice. Let's recap the most critical issues covered in this eBook.

### **Concept and Detailed Design with 2D Geometry**

Concept design and the early steps of detailed design are both natural fits for 2D geometry. In both, designers and engineers need to explore alternatives and options with minimal commitment. In these cases, designing with 2D geometry represents the right amount of effort.

### **Designing with Drafting Tools**

One way of developing designs with 2D geometry is with Drafting Tools. Designers and engineers must create and modify 2D geometric entity one at a time, undermining efforts to iterate and explore options and alternatives.

### **Designing with Feature-based Sketchers**

Another way of developing designing with 2D geometry is with Feature-based Sketchers. While they can drive intelligent change in 2D geometry, their performance degrades with large number of entities, which is common when a cross section from an existing design is used as context. Furthermore, a sketcher's need to fully define a section hinders a user's ability to explore designs.

### **Designing with Direct Sketching**

A new alternative to Drafting Tools and Feature-based Sketchers is Direct Sketching. They use local temporary

assumptions to power smart changes to 2D geometry without performance degradation, yet offer the means to embed global design intent as necessary.

### **The Value of Swift and Smart 2D Design**

For companies, the main advantage of swift and smart design with 2D geometry is simply getting to a better design. The benefits of that, in turn, can range from capturing market share, increasing profits and more. But by and large, outcomes related to better designs are all positive. For individuals, it stands a means to cut back on working late and on the weekends because it offers a shorter path to design completion.

Today, many companies face an ugly compromise on whether to use Drafting Tools or Feature-based Sketchers to design with 2D geometry. But with Direct Sketching, swifter and smarter design with 2D geometry offers a very real opportunity to produce better designs and reap the associated benefits.

*Underwritten in part by Siemens PLM, all concepts and ideas developed independently, © 2013-2014 LC-Insights LLC.*



Chad Jackson, the Principal Analyst of Lifecycle Insights, is a recognized authority on technologies that enable engineering, including CAD, simulation, PDM and PLM.

Contact / Follow:

