Applying mass customization & visualisation successfully

APPLYING MASS CUSTOMISATION & VISUALISATION SUCCESSFULLY

Considerations regarding the use of product configurators, product visualization on a large scale and integration into your IT landscape

---

Mass customization can be defined as enabling your customer to tailor a product to their own liking and have it manufactured with a back to back order. It is common in many industries including furniture, jewellery, automotive, clothes and shoes.

Why mass customization? The most common reason is competitive advantage; being able to offer your customer their own unique products. Other benefits include reduced logistics, waste, storage and shipping costs along with increased accuracy, control regarding product information and ultimately customer engagement.

Whilst customisation is not a new concept, until recently, there was no technology available to properly structure the product data, visualise it and automate the selection process. Until now, sales staff would need to talk customers through the options to create the sales orders containing the "configuration". There are at least two problems with this process; customers often have to rely greatly on their imagination to have some idea of how the end product will look which may lead to a mismatch of expectations and 2, a lot of costly mistakes are made in creating the sales and purchase orders such as invalid configurations, omitted items and misinterpretations of the customers' wishes.

In this whitepaper, we discuss how mass customization can be automated with advanced product configurator software and work towards a best practise model considering commercial, visual, and logistical considerations.

To be able to offer (automated) mass customisation, three challenges need to be met:

1. **Selection:**
Which products and which product options can you offer to enable your customers to confidently customize products?

2. **Visualisation:**
How can you show your customers what their "made to order" product will look like, whilst keeping costs at acceptable levels?

3. **Flawless supply chain management:**
How do you ensure that customers cannot create invalid product compositions and ruling out mistakes in the purchasing / manufacturing of end products?

This paper will conclude with a short recommendation to get you started on a future first model.

1. ***Selection:***

Which product s and which product options can you offer?

One of the challenges of mass customization is managing the sheer number of options that you can theoretically offer. For example, in furniture, the number of product configurations can be huge. To give you a flavour of the scale of the issue, let's take a real-world example of a simple sofa. Here is what you can configure:

- Sofa type: 2 seater
- Fabric groups: 2 options (fabric and leather)
- Quality/price groups: 5 options per fabric group
- Colours: 20 options per quality group
- Stitch colours: 10 options
- Feet types: 4 options
- Feet colours: 8 options

This is a relatively simple example and at the lower end of what is possible, however if you multiply these options, there's many possible configurations:

2 x 5 x 20 x 10 x 4 x 8 = **64.000**

It is important to realise that this whitepaper does not address challenges which you might or might not be facing in the future. These challenges lie in the here and now, with actual products that you might already be offering to your customers. In other words - you are already selling these products and your objective is to simplify the navigation and complexity of your product offering.

So how do you get your business ready for (online) mass customization?

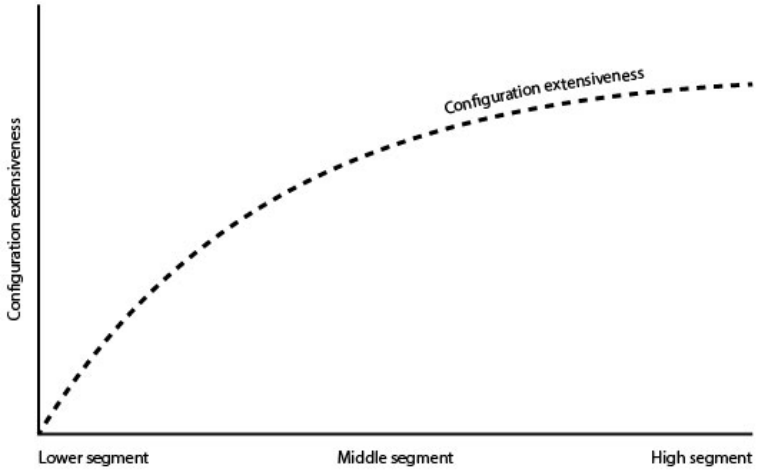
The first thing you need to have is a powerful product configurator that can handle all those product configurations including all possible exceptions. It will help you to manage your product offerings to your customer, because looking back at our earlier example- you do not want to show 64.000 different product variations of a single model on your webshop and especially not in your store.

Once you have gained access to a product configurator, you can start creating configurations. However, there are still some considerations to be made as to "who is the intended end user of the configurator?" and "how elaborate should configurations ideally be depending on product segment?". Your considerations should be tailored to your specific business model, but let us offer you some basic guidelines based on our experience.

1.1 Basic product segmentation

Common marketing theory suggests that a trade-off exists between product value and the effort a customer is willing to invest to obtain it. For instance, if you're going to buy a new car, you are likely willing to travel to the next town (where the dealer is located), whereas you are less likely to do so in case you desire an apple (the fruit, not an apparatus in this instance). Generally, a customer is willing to put more effort into higher segment products. In some instances, they even expect it, as they assume that higher segment products offer more breadth (which require more of their effort to make the choices) as opposed to lower segment products. You would be more than willing to take the time to go through a configurator to purchase your new car, but might feel less willing to do so for an apple (again, the fruit). We recommend that the extensiveness of your configurations should depend on the segment the product is in, to ensure the best fit with customer expectations. Undoubtedly arguments might be found to deviate from this basic model for certain products due to commercial or logistical considerations, but overall the product segmentation is a solid starting point.

1. Lower segment products – small product configurations
2. Middle segment products – limited product configurations
3. High segment products – larger product configurations



1.2 Defining end users

Before you start to make product configurations you need to determine your target (end) user groups. We can distinguish 3 groups:

1. Internal users
2. Consumer users
3. Business to Business users

1.2.1 Internal users

When you build product configurations which you intend to use only in your own store(s) and which will only be used by your own personnel (sales people, order entry, etc), you can create more complex product configurations. This is based on a trio of assumptions: (1) the users are acquainted with the products they are configuring, (2) you can provide some explanation about the product configurations that you have made through training and memo's and (3) these users will use the product configurations frequently and they are likely to learn to use them correctly. From a commercial perspective, you may want to offer internal users a more complete range of product options compared

to the configurations you offer for consumer users for two reasons: (1) for certain options consumer users might lack the required knowledge to make a choice and (2) being able to offer additional options only in your stores is a good way to ensure traffic to those stores.

1.2.2 Consumer users

If you want to offer your product configurations to the consumer through web or in store interfaces, it becomes a challenge. In the first instance, a consumer may use your product configurator for the first time. Overview and clarity are required to make the experience enjoyable and understandable. Secondly, they may be impatient as they may just be browsing and want to have a quick overview of the possibilities. The content needs to be delivered fast and you only get one chance to make a good impression. Thirdly, you want to avoid complex products and/or asking customers to make any product configuration decisions which are not common knowledge. Therefore, you should apply other guidelines for product configurations that you offer to consumers. Less options provide a sense of overview and increases confidence that he/she is making the right choices - co-creating their ideal product. Less options offer a better performance of the product configurator in terms of speed, which contributes to the joy of customising products.

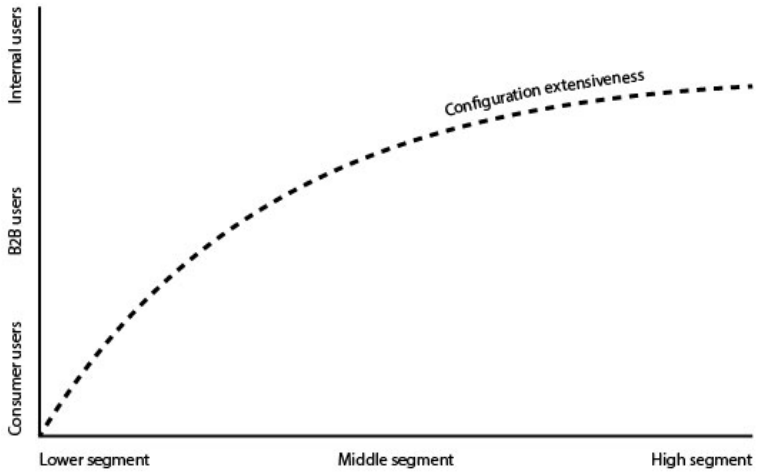
1.2.3 Business to business users

If you are a manufacturer or a supplier, you may want to share your product catalogue including your digital assets with your customers. In the first place that helps you sell more products and in the second place digitalising the order process increases efficiency and will reduce incorrect ordering. Although you offer your products to users who are experienced in your line of business, you still need to be careful with the number of options you provide. If it takes too long for a sales person to 'walk through' your product configuration, then there is a risk that they will avoid using your catalogue and they might sell a similar product from your competitor.

1.2.4 Mixed user groups

It is quite likely that your configurations will be used by more than one of the user groups described above. Creating separate products for each user group is not an option as it will lead to an administrative and logistical headache. Your ERP or PIM application requires a "next-gen" product configurator that is either able to publish multiple versions of the same product for different purposes (users), or allows you to build multiple configurations which will still lead back to compositions of correct SKU's (i.e. a certain screw used in 2 completely different product configurations, is still the same screw and should thus be ordered under the same SKU regardless of the configuration it originates from). A proper product configurator should enable you to do both which then enables you to tailor the best fit for each user group from a single configuration, simply by using defaults and publication settings for the different user groups.

To elaborate on our previous "configuration extensiveness graph", we've added the various user groups along the Y axis in the graph below.



1.3 Commercial considerations

Product segmentation and the definition of your intended users provide a base model to define the extensiveness of your configurations. However, one might argue exceptions can be made due to commercial / marketing considerations. You might think of one of your best-selling products which falls in the lower end of your product range and for which you still want to create a rich configuration for your webshop (consumer users). Or, a high-end product (configuration) intended primarily for internal users for which you want to provide only a limited number of options because of logistical reasons or purchasing benefits. These are justifiable exceptions. There is no right or wrong if configurations are created with a clear vision and goal. Most importantly though, one must always keep the math in mind.

1.4 Always do the math

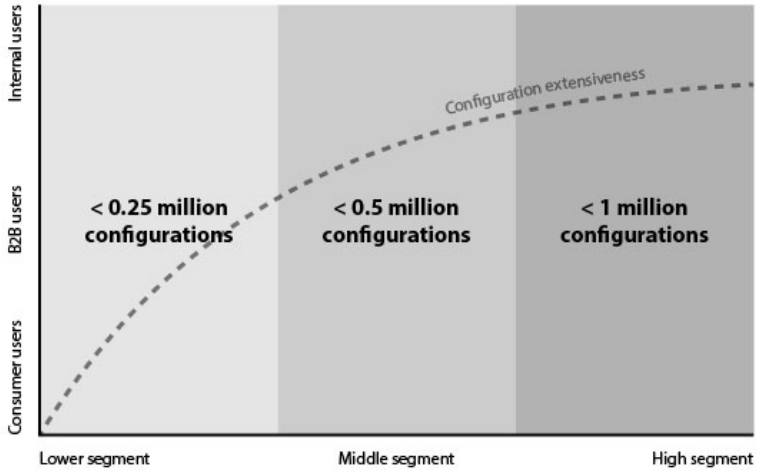
Earlier on in this chapter we've included a simple calculation of the number of configurations a simple product would amount to; just 6 options and a total of 49 choices resulted in 64.000 possible outcomes. Add just one additional option with only 10 choices to this model and this number would increase to 640.000 configurations.

Imagine when your product configurator application would have to handle hundreds of these product configurations. Then, imagine when you would add imagery to these configurable products with an average image size of just 0.5 MB. Finally, imagine what your application must process when multiple users are simultaneously clicking through the millions of options and terabytes of (visual) data. Even the most powerful applications running on the fastest servers can only handle so much until performance and/or storage becomes an issue.

The point we're trying to make is that you should tune your configurations to the mix of commercial interests, intended users and product segmentation, but most importantly; never lose track of the number of possible outcomes (i.e. product configurations).

So, what number is a reasonable one? As there are so many variables to consider (i.e. server, power of the application, imagery, number of products, etc), you are the best person to answer that question yourself. However, based the current state of accessible technology and our experiences, we propose sticking to the following ranges:

Class	Total number of possible configurations:
Consumer use and/or lower segment products	< 250.000
B2B use and/or middle segment products	< 500.000
Internal use and/or high segment products	< 1.000.000



1.5 Considerations in choosing your product configurator application

A growing number of product configurator software applications are available and there's no clear-cut definition of what features an application should contain to be called a product configurator. So how do you decide which application suits your needs best? There are some aspects to consider for example:

- How generic is the product configurator: can you create all desired configurations in this one application, or does every model require some custom development by the vendor?
- Is the configurator a stand-alone solution or does it have out-of-box features and/or API's to truly integrate into your business processes (i.e. can you create sales orders from the built configuration with all the right logistical items / SKU's?).
- Does the application only offer a solution for my back-office to build configurations, or is it just a front-end solution where customers can choose (limited) configurations, or does it offer both?

Additional considerations to consider which may make your processes and use of product configurations a lot more efficient and enjoyable:

Re-using options

Does the product configurator enable you to re-use options and sets of options? Creating a product configuration can be a lot of work. It's highly likely that a large portion of your products share common parts or options which you do not want to create from scratch for every product configuration you build. Sophisticated product configurators offer features that enable you to re-use options or even sets of options which you have created for another product configuration. I.e. when you have multiple products which share a common set of materials, you will want to be able to create a set of these materials and simply add to the next product. Picture again our example 2 seater sofa which offers various fabrics to choose from (2 fabric groups and 5 quality groups with 20 options each, resulting in a total of 200 possible choices). Obviously, there's much efficiency to gained if you're able to just copy this set of options to the 3 seater model as well.

Also, be sure to check to what extent you can re-use option sets while retaining control to then deviate from these options in that specific instance. For instance: let's say the 3 seater model should not offer

all of the fabrics which are offered for the 2 seater model. Does your application allow you to still re-use the set and then subsequently create the required exceptions?

Grouping options
Some product options are vast in number and categorised into smaller subsections, which makes displaying them a challenge - no matter which user group you have in mind. Take for instance the fabric / colour options in the before mentioned sofa with (2 fabric groups x 5 quality groups x 20 colours =) 200 options, divided in (2 fabric groups x 5 quality groups =) 10 subcategories. Without a single overview of all the options, you might be able to imagine that going back and forth in these categories to view the eventual colour options will be confusing. When choosing your product configurator software, be sure to check whether features exist to handle this properly.

Price management
You can image that price management on these vast numbers of configurations can be a daunting task. How daunting really depends on the structure of your products but, mainly, the features your product configurator application should offer. Not every option in your configuration may have a price consequence but, generally, a lot of them do. Commonly, there are at least a couple of price consequences on the level of single options. For example, size or material (e.g. some sizes and/or materials are more expensive than others). Where it becomes more complex, are the (very common) situations in which a combination of options will define the price. For instance, when the combination of shape and material define the price of an option or configuration. Being able to efficiently manage thousands of prices within your configurations is something very few applications offer, yet undoubtedly you realise the importance of these features by now.

Business rules
Usually, product configurations are not so straight forward in a sense that each option can be combined with all other options without restrictions. Moreover, sometimes price increments depend on a combination of choices made and not just one particular option. There are a lot of rules, exceptions, and price consequences a product configurator should be able to handle. Let us provide some examples:

- If option "A" is chosen, then option "X" should not be available later in the configuration
- If option "A" is chosen, then option "Y" should be added in the configuration
- If option "A" is chosen, this choice should be automatically applied to other components in the configuration
- If option "A" is chosen, option "Z" should have a price increment of X
- Etc.

We don't intend to discuss all possible rules and exceptions in this whitepaper as there are countless of situations to consider. We merely want to point out that you should carefully check which features your product configurator offers in this respect. A product configurator that offers limited or no features to apply complex business rules to your configurations would greatly limit the usability and would get you stuck in no-time.

2. ***Visualisation***

How do you show your customers what their made to order product will look like?

Regardless of the techniques used, the result must be a complete image representing the product in the correct configuration as selected by the user/customer. Traditional photography will often not be an option in case of mass customization. Take our sofa for example; that would require you to have over sixty thousand configurations manufactured, photographed, moved, stored and (hopefully) sold! Digital rendering is easily 50% to 80% cheaper than traditional photography for even the relatively simple products. For average complexity products like our sofa example, this might rise as high as 99%.

There are a many different approaches to this even when using computer generated images (rendering) exclusively and each method has its own pros and cons. It is not a one size fits all. One thing they do have in common; they all require the creation of a 3D model to start with.

What are 3D models?
3D models (for these purposes) can be defined as digital representations of physical products modelled in 3 dimensions. 3D models can be viewed from all angles and distances within a digital environment and, furthermore, they can be interacted with (moving them and/or changing them).

It's important to note that 3D models can have a wide range of applications, which -in the near future- will greatly increase. 3D models can be viewed in a traditional 3D viewer as such, but this same model can also be used in product development and manufacturing processes, it can be applied in augmented and virtual reality applications as well as interactive 3D product configurators, room-planners and even product videos. Thus, investing in 3D content sets you up for a wide range of currently and future available possibilities. For this whitepaper, we focus primarily on one specific application; your 3D models may serve as a source from which rendered imagery is created.

What are renders?
A render is a high fidelity, high-quality and very realistic digital image of a 3D object. 3D renders can complement traditional product images and may even be used to completely replace it.

3D rendering enables you to show customers the full extent of your product ranges with all possible options and in all possible combinations. This may sound logical, but up till now customers have had to base their decisions on limited information while buying products as it was just not possible to show all variations. Early adaptors can reap huge benefits as developing these digital assets takes time which competitors cannot easily catch up.

Advantages of 3D renders vs traditional photography
As products, often are available with many options, creating (digital) assets such as product photographs and videos is challenging, costly and time consuming. Often, this results in a lack of (proper) imagery of products while the importance of images is becoming increasingly vital, especially in a web environment. How can 3D fix this?

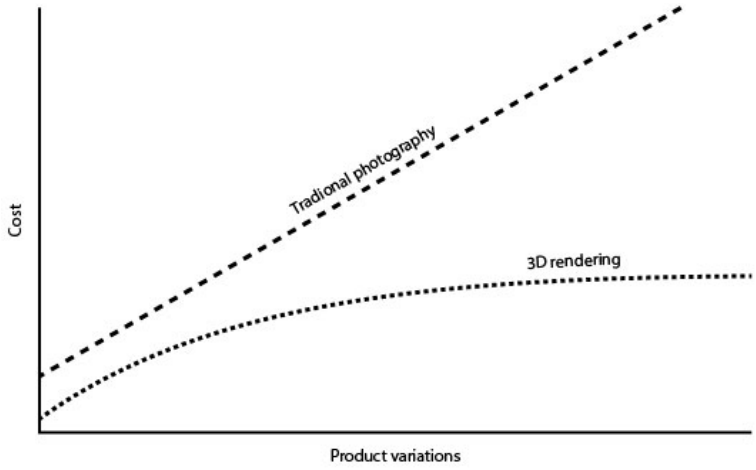
- Products do not have to be manufactured to develop images
- All options can be visualized and can be displayed in all possible combinations
- Once the models are created applying different fabrics and different colors to the models is easy
- No need to hire a photographer
- No studio space is needed (rooms can be created just the same as the models themselves). Apart from saving costs, this enables a wider variety of spaces

- No logistics (and consequential costs) are involved, while traditional photography requires products to be moved to and from the photography location
- Digital products require no physical space (manufactured goods need to be stored, displayed, and sold)
- No accessories are required to "liven up" the picture, as they can also be created digitally (this also affects purchasing, storage and logistics costs)
- 3D models are omnichannel. They can be displayed in stores, in brochures, on the web, in videos, in apps, augmented reality, virtual reality, etc.

The costs involved in traditional photography and 3D modelling / rendering obviously depend on the product. For instance, more portable products can usually be photographed more easily and with fewer logistical costs involved than bigger products. Also, recreating products in 3D with hard surfaces and simple shapes is generally a lot easier compared to products with complex (curved) shapes and soft materials (such as fabrics). There are a myriad of variables to consider which define the costs of photography and 3D modelling / rendering, and therefore a catch-all cost comparison is impossible to make. If you intend to create a single or just a limited number of product visualizations the costs of creating 3D imagery might not be that much lower or, in some instances, even higher compared to traditional photography. However, as the number of product configurations increases, the cost of traditional photography will increase rapidly, while the cost per 3D generated image will diminish with each additional visualization. In short, when visualizing large numbers of product configurations, 3D imaging might save up to 99% compared to traditional photography.

Let's take the before mentioned sofa as an example. Hiring a photographer to shoot a single image of one specific configuration would cost a total of €500 (product manufacturing/purchasing cost, logistical costs, studio rent and hourly rate of photographer included). Let's assume that creating a rendered 3D image of this very same product would cost about €400 (product manufacturing/purchasing cost and hourly rate of 3D modelling artist included). Not much to be gained so far, right?

Now, let's imagine that we want to create visualizations for all 200 available fabrics in which this sofa can be configured. As you would have to have each configuration manufactured, transferred, stored and photographed you can imagine your cost for each image would increase almost linearly (200 x €500 = €100.000). In 3D, however, these 200 fabrics can simply be photographed or scanned based on fabric samples and then easily applied to the base 3D model. Assuming the addition of each fabric and the generation of each additional render would cost around €10,- per image, your visualization costs would be much, much lower: 1 x €500 + 199 x €10 = €2.490,-. That comes down to a decrease in visualization costs of almost 98%!



Techniques
Now that we know the financial advantages of 3D models and rendering over traditional photography, let's look at the technical options that we have:

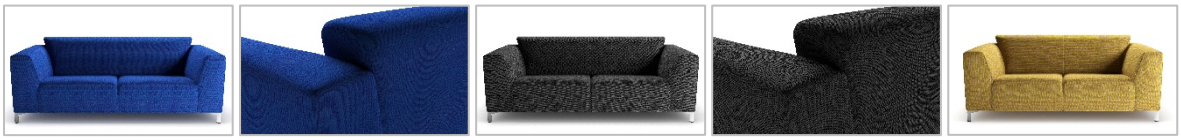
1. Pre-generated Images
2. Composited Images
3. Live rendering (Dynamic on Demand Images)
4. Client-side 3D
5. Hybrid Client-side 3D and Live rendering

2.1 Pre-generated images

The simplest approach is always to just pre-generate all the render images that you will need in advance and just deliver the appropriate image for a given configuration.

This approach can be practical when the total number of configurations available is in the order of 10.000 or less. If you make assumptions about how long each image will take to render and calculate how many combinations you need, then you can get an idea for how much time you might need to wait to get it all done.

Even when using pre-generated images, you are still going to want to automate the rendering process.



Advantages:

- Quality: pre-generated render images typically provide the highest quality as the generation time is not usually a primary concern (although it is a factor) since the customer will not be exposed to it.
- Simplicity: as the images are pre-generated they can be handled like regular images, filed, and indexed and retrieved on demand by the web application based on their naming convention. The images can also be cached in a CDN (Content Distribution Network) for fast display/delivery.
- Automation: with the right tools, the pre-generation process can be automated.

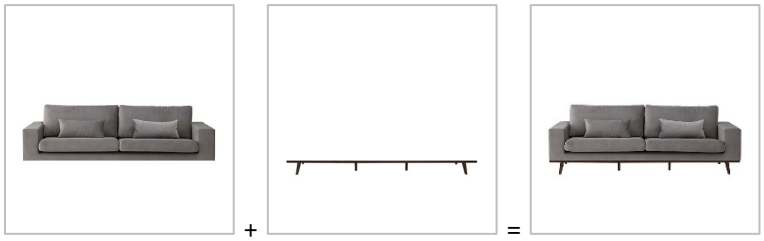
Disadvantages

- Precomputation: everything needs to be done in advance and if the number of combinations is large this could involve the use of an expensive render farm of computers as the rendering process itself takes heavy duty computing power.
- Lack of Flexibility: typical configuration applications can quickly run into millions of possible configurations at which point it becomes impossible to pre-generate the images in a cost effective and timely manner, forcing the configuration options to be artificially restricted.
- Wasted Resources: it is possible that many configurations that will never be seen by a customer will be generated. Since it is not known which configurations will be selected, all of them must be generated in advance, whether they will be used.
- Maintainability: if a change is made to the content, all images may need to be completely regenerated. This can be particularly problematic when a small change is required that effects many configurations.

2.2 Composited Images

Probably the most popular approach used today for configurable products is compositing. Here, rather than pre-generating all possible combinations, you generate individual elements which have been separated based on their configurability. These elements can then be combined in different ways to make the required configurations.

The goal with composited images is to break the problem down into pieces where the combinations of each piece works out small enough to handle with precomputation. It's essentially a divide and conquer approach and basically it just moves the line where the number of combinations is practically enough to pre-compute. To understand why this works we'll look at a typical example.



Let's take the Sofa mentioned earlier. If we brute force it (using the pregeneration technique) we will have to generate the following quantity of images:

- Type sofa: 2 seater
- Fabric groups: 2 options (fabric and leather)
- Price groups: 5 options
- Colours: 20 options
- Stich colours: 10 options
- Type feet: 4 options
- Colour feet: 8 options

Pre-generation
2 x 5 x 20 x 10 x 4 x 8 = 64.000 images

With plenty of hardware, lots of time and a reasonable rendering time this will probably take a couple of weeks or even months (i.e. if rendering time is just half a minute per image, it would take one machine (64.000x0,5min/60min/24hrs=) 22 full days to pre-generate all images).
What does it look like when we take a compositing approach?

Compositing
(2 x 5 x 20) + 10 + (4 x 8) + 1 = 243 images
The +1 at the end is for the image containing the static elements which don't change between configurations.

Now these are not complete images of the possible configurations, they are fragments of it and you will need a powerful application to combine these into the actual configuration shown in a client (e.g. browser). You can see however that this approach can significantly reduce the number of images needed.

Compositing can however get tricky or sometimes impossible when you must deal with geometry changes (e.g. the actual form of the model changes). This can only really help if you start taking 2.5D approaches and compositing with depth considered.

Transparent objects make compositing even more tricky, especially when their transparency can be variable since they will potentially reveal objects or colours that were not originally included and so give incorrect results. For example, a dress made from two basic components: (1) the base model made of silk in 10 different colours and (2) several details that drape over the base model made of transparent lace in 5 different shades of white. In theory, one could generate separate images of the base models and the lace components, and subsequently composite the images (placing them on top of each) other depending on configuration choices. However, that wouldn't lead to the desired results at all. Obviously, the lace itself looks different when placed on top of a green or a red dress due to its transparency. In this scenario compositing wouldn't work and, instead of generating (10+5=) 15 images, you would have to generate images of all (10x5=) 50 combinations.

However, for the right use cases compositing can still be a great option - you just need to make sure that your products feature limited or no geometry changes and do not contain any transparent components of which the appearance depends on (interchangeable) components behind the transparent object. Here's a quick summary.

Advantages:

- Flexibility: can handle much larger numbers of configuration combinations than pre-generated images since image fragments can be combined in many ways.
- Quality: when using compositing the quality of image fragments can approach that of pre-generated images (essentially, they are just pre-generation at a different level). There are however some quality issues as mentioned with edge artefacts which need to be handled carefully.

Disadvantages

- Quality issues: there is always a compromise since edges must precisely fit together.
- Geometric configuration issues: visibility needs to be considered (i.e., whether something is in front of or behind something else). This requirement creates complex issues for example when an object is both in front of one object and behind another object at the same time.
- Missing phenomenon: since the compositing process cannot account for interactions between image fragments there will be missing effects. For example, indirect colour tinting, reflections of configured items and in some cases shadows of configured items.
- Server-side resources: compositing that takes place on the server-side is resource intensive and can require significant processing power relative to just serving web pages and images.

2.3 Dynamic on demand images (live rendering)

In this approach, no pre-generation is performed. All 3D content, geometry, textures, materials, and lighting reside on servers. When a configuration is selected the rendering software dynamically renders the required image real-time and sends it to the customer.

Using this approach each configuration is rendered entirely from scratch and so it can account for all possible configurations without the need for complex compositing or extensive pre-generation of content (i.e. image generation is done only on demand). Often a web application will sit between the rendering software and the customer to manipulate or manage the generated images in some way.

Advantages

- Flexibility: this method can account for virtually any configuration a customer might chose since there is no precomputation. Additionally, content can be modified and changed at any point and customers will automatically see the new/adjusted configuration.
- All phenomena: since each configuration is rendered on demand, all effects such as indirect colour tinting, reflections, shadows, etc will be accounted for.
- Quality: can easily approach pre-generated images however quality desired is tuneable based on the response time acceptable for the customer's request and the amount of resource/processing power that can be dedicated to this service.
- Maintainability: since the generated images are never stored but just sent directly to the customers it is possible to modify the content on the server to incorporate product changes and updates without the need to pre-generate all the content again. All customers would see all changes immediately.

Disadvantages

- Server-side resources: unlike conventional web-sites, producing server-side, photorealistic images require specific server hardware and a significant amount of computing power.
- Longer response time: since images must be rendered from scratch they can take longer to produce than simply serving a static image. However, it is worth noting that compositing based systems can also take an amount of time to produce results.

2.4 Client side 3D

In recent years an increasing number of companies began using 3D directly in the browser with technologies such as WebGL, Three.js and Unity3D. This is a compelling area of development since it pushes the 3D hardware requirement to the end user and removes the need for brands to host this capability on their customer's behalf. There are however some challenges.

Pushing the hardware requirements to the end-user is a good thing, if they have decent hardware. It's getting easier to assume this with recent iPads and Android devices having very capable (3D) processing capacity, and in addition, most laptops and desktops now having enough grunt to do some level of 3D, thanks to WebGL.

Looking at projects we have done so far, the data sets can be a challenge especially if you want to offer it to consumers (i.e. let the consumers configure a 3D model on the web/mobile). Many 3D models comprise around 5 megabytes of data (when compressed) and a couple of hundred thousand of polygons. These factors mean that a lot of information needs to be moved to the client side before the first image appears. With your average internet connection, this implies a waiting time of approx. 8 seconds. This raises a lot of performance / acceptability questions based on an important metric we often deal with and like to call "Time to First Image", which defines the time it takes from the first moment an image is requested to the moment it is returned and displayed. This is important for all image generation methods but becomes a significant factor with client side 3D, because images cannot be generated until the data is transferred to the client. You can influence the user experience by splitting the model and the materials so the files are smaller and it can be download asynchronously.

If the consumer is aware that they are working in a 3D product configurator, this is not a problem as performance expectations are lowered because they know they're dealing with a complex environment. But once you incorporate this same configurator into another environment (e.g. a product page of a webshop) acceptance is a different story as consumers base their expectations on average webshop load times. In other words; with the change of the primary environment (from 3D configurator to webshop), consumers suddenly no longer show courtesy towards load times.

Because of the limitations you have on file size (the better the quality, the bigger the file and the longer it takes to download), client-side 3D technologies are limited in the quality of the models they can visualize. Some client applications have managed to show some very impressive content in WebGL but, in all fairness, it still doesn't have the quality that can be achieved with photorealistic rendering. For many applications, however, it might be good enough.

Advantages

- Interactivity: once 3D content is downloaded, provided the user has reasonably modern hardware, content can be freely navigated at high frame rates.
- Modifications: configuration changes can potentially be made immediately once the content is downloaded without making requests to the server.
- Less server-side resources: offloads computing costs associated with the system to the users' computers rather than the servers.

Disadvantages

- Download time: all source 3D content used must be downloaded to the users' device before it can be viewed. This can significantly increase the time to first image and the size of these downloads can be considerable depending on the quality of the content.
- Image / model quality: quality that can be achieved is limited. While there are techniques to obtain quite high quality results, they need pre-generation that would negate the ability to configure the objects easily.
- Hardware requirements: users with older hardware that lack 3D acceleration is likely to receive an unsatisfactory experience.

2.5 Hybrid: client side 3D and live rendering

That's a long title but in short, what we do here is combine client-side 3D technologies with image generation. The idea is to try and get a 'best of both worlds' approach, where interactivity is high but high quality can also be achieved when needed.

Typically deploying this approach would involve using two sets of 3D content, one which has been simplified and compressed for client side 3D and the other full detail content used for high quality rendering. These would be two versions of the same content and be setup with the same structure and positioning.

When the user initially visits the site, the low detail 3D model would be downloaded and viewed, allowing configuration options and views to be selected easily and providing some level of feedback even if the quality isn't very high. Once configured the customer could either specifically request a high quality version of the rendering, or alternatively this could be requested automatically when the user stops changing the configuration.

Advantages

- Interactivity: during configuration response can be immediate since everything occurs on the client side and the user can keep interacting, even if at a lower quality, if there are any issues on the server side.
- Quality on demand: rather than use a large amount of resource to get full quality always, this approach allows you to serve the higher quality results at specific key points in the purchase process, for example prior to finalising a sale.

Disadvantages

- Download time: like client-side 3D the time to first image can be quite long, even with a simplified model. However, once the first download completes subsequent interactions can be much faster. During this time, the server-side counterpart can also be spun up.
- Limited quality during interaction: while the customer would request high quality on demand, the quality during interaction will be limited to that which can be achieved through WebGL or the chosen client-side technology. However, it might not be a fair comparison as the first three models feature no interactivity at all.

3. ***Flawless supply chain management***

When you've managed to create proper product configurations and you have chosen a suitable visualization model, you have worked out the front-end side of mass customization. That means the hardest part of the job is done, if you have the proper application to support these processes in the back-end (transaction administration, purchasing / manufacturing, warehouse management, logistics, etc). Enabling your customers to freely configure products without automation in your back office would likely lead to disaster, as unrestricted customization may lead to invalid/impossible configurations. Updating product configurations and prices would take enormous amounts of time and everything that comes out of the configurator needs to be processed manually resulting in giant heaps of work and lots of incorrect orders with due costs. We touched on most of these items in the first chapter however, these items were primarily discussed from a rather commercial perspective. Now we would ask you to consider these same items, only from a more logistical point of view. At which (configuration) level should your logistical items (SKU's) be defined in order be able to automate and connect all other processes that follow a successful configuration converts to a sales order?

All transactions should be processed directly and digitally leading to clear and flawless sales and purchase orders, which contain the complete configuration breakdown including all separate components with correct SKU's and specifications. You need to be able to digitally transfer purchase orders directly to your supplier / manufacturer and track movement and stock on individual components which serve as input for warehouse management and delivery planning and can be communicated to your customer.

4. ***Get started!***

Intoduction to iOne business software

In this whitepaper, we've discussed the implications of automating mass customization in your business based on three main area's: 1. building configurations in product configurator applications, 2. visualisation of your product configurations and 3. the integration into your IT landscape to properly process all the data involved. By now you will have realized the complexity and the vast number of variables to consider. You will likely also realize the importance of choosing the right software to enable (automated) mass customization in your business.

We have kept this whitepaper as neutral as possible to provide unbiased information with the goal to help all readers in decision making processes with regards to this subject. The experience, information, and considerations which we have shared with you all stem from the fact that we have spent many years working in this domain and building a software solution that truly enables you to apply, and automate mass customization and visualization. Therefore, we would like to take the opportunity to introduce our application "iONE". iONE is a business software application which offers an advanced and very comprehensive product configurator, among many other features. Although primarily used in the furnishing industry, its generic nature makes it perfectly suitable for a large range of other industries as well. iONE's product configurator is developed with a philosophy that configurations for virtually any type of product should be supported through a clear, generic, and very powerful configuration engine in

which users can build their own product configurations. iONE also features an extensive business rule engine capable of handling a large variety of rules, exceptions, price consequences and even promotions, fully integrated into the product configurator. It also supports various visualization options including traditional imagery, composited renders, live rendering, and even full 3D product configuration. Furthermore, iONE enables you to publish your product configurations in various ways and on various environments such as a fully web-enabled product configurator which can be integrated into all kinds of ecommerce solutions. iONE can be licensed as a complete and standalone business solution which includes features such as warehouse management, purchasing, point of sales, relations management, delivery planning and service management solutions, or simply as a product configurator on its own.

For more information about this whitepaper or iONE360 business software, please contact us directly on www.ione360.com/connect