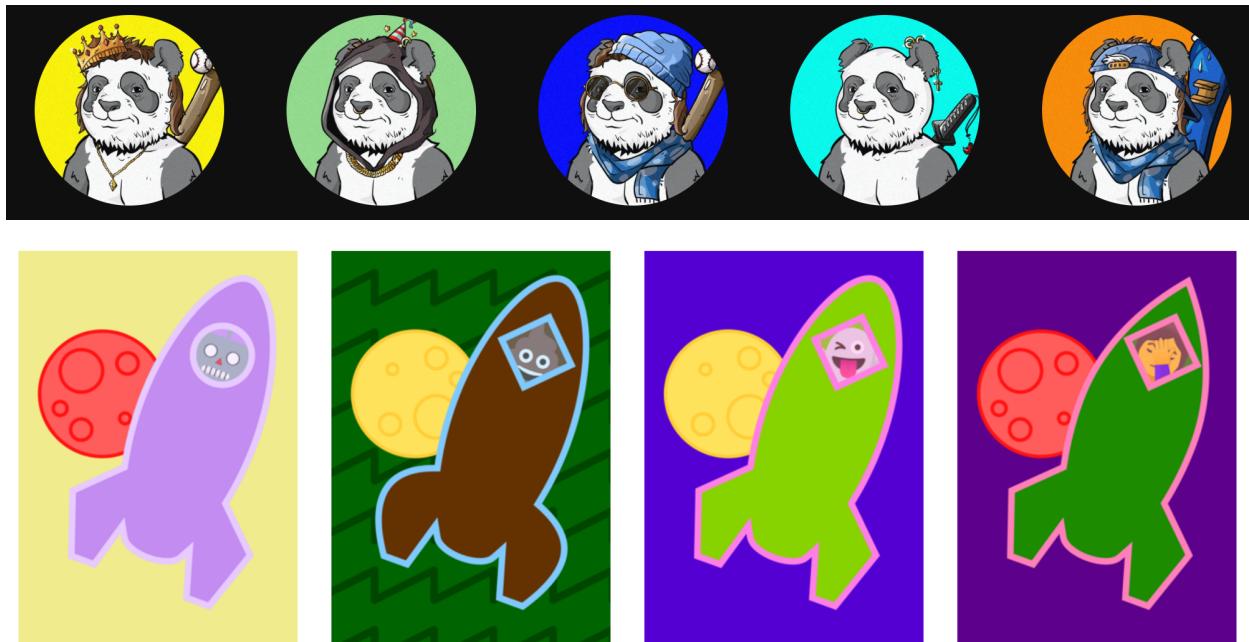


Description of NFT Engines and Processes:

At Prakash NFT Co. (also known as NFTech) we have built a series of tools and processes to help create a series of NFTs in an hyper-efficient way from inception to design to release. There are multiple “NFT Engines” or code bases, and processes that we have engineered and designed in order to accomplish this:

Instead of creating one-off NFTs from pieces of art or graphic designs (like many of the NFTs we see on OpenSea, Rarible, or the like), we like to make use of our “NFT Engines” and certain processes that decompose an image into ‘traits’ or ‘features’. We then use one of our “NFT Engines”, more specifically the “Prakash NFT Auto-Compose Engine”, to assemble these traits (randomly, with the ability to specify how often a variation of a particular trait appears e.g. smile 50%, frown 30%, grin 10%, tongue out 10% of the time for the ‘*mouth trait*’). This allows for us to create a total number of permutations or combinations that reach into the millions and billions, while creating rarity among the created NFTs without burdening the artist or graphic designer with creating too many assets. From these millions or billions of possible combinations we generally will only mint 1,000, 10,000, or sometimes 20,000 NFTs depending on the need, market demand or specifications. A great example of this is PandaDynasty (<https://pandadynasty.io/>, <https://opensea.io/collection/pandadynasty>) and a simpler example is LilMoonRockets (<https://lilmoonrockets.com/>).



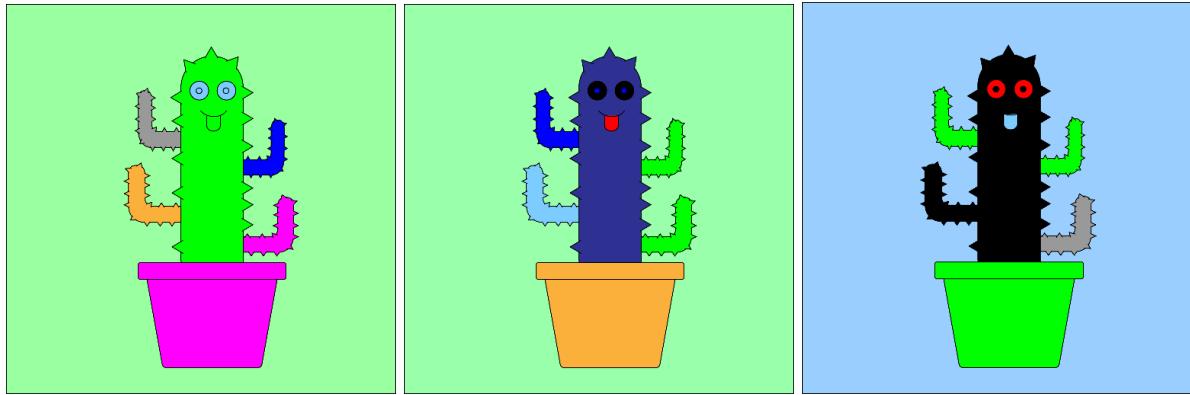
We often like to take this rarity one-step further and make *trading cards* out of these images described above, allowing for descriptions and other details about the image or artist/graphic designer who created the art work or details of the cause that some of the proceeds of the sales will be going to. The cards below are part of the *ZodiaCards NFT Series*, where 20,000 NFTs were created out of over 85 million possible combinations.



1. Creation of an “NFT Matrix” that breaks down the image to be created into a series of traits and variations of the traits (and their respective rarities i.e. how often they will be selected by the random selection algorithm of the “Prakash NFT Auto-Compose Engine”). E.g. Trait = Hat: Baseball Cap Blue 10%, Baseball Cap Red 10%, Baseball Cap Gold 5%, Black Top Hat 25%, No Hat 25%, Do Rag 10%, Wig 10% and Crown 5%.
2. Creation of the traits as assets in the specific (x,y) position for the NFT Engine to place or layer together to generate the resulting image. E.g. The Hat Trait would be offset in the y-direction so when the NFT Engine assembles the image together - it will always be in the right place and layered after the face/body is placed first.
3. Run the “Prakash NFT Auto-Compose Engine” with all the assets created - again only a small portion of the total possible combinations/permuations will ever be created. The rarity specifications will allow the creations of “common, medium, rare, super rare, ultra rare, etc.” NFTs automatically.

Visual Description of the Above:

These “Cactus Club” images/NFTs below were generated via the Prakash NFT Auto-Compose Engine as part of our testing of the code and as an easy way to describe how the process works:



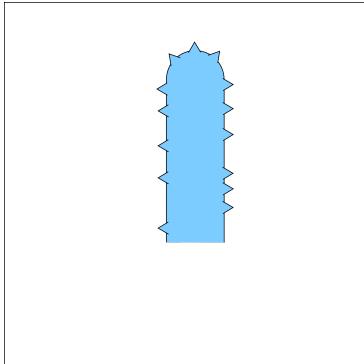
These images/NFTs were composed from 8 different traits: 1. Cactus Body, 2. Pot, 3. Left Upper Arm, 4. Left Lower Arm, 5. Right Upper Arm, 6. Right Lower Arm, 7. Tongue/Mouth, 8. Eyes/Pupils.

Each of these traits don't have asset variations but 12 color variations (per trait) instead (as it was a test/illustrative example). In other words, for example Trait #7 Mouth/Tongue does not

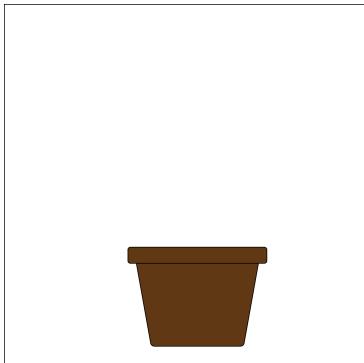
have different mouth variations e.g. smile, tongue out, frown, grin, etc. but just different color tongues. A real series would have some combination of asset and color variation for each trait.

Each cactus is composed of the 8 traits mentioned above, and the below images illustrate the fact that each trait and asset must be in the **proper x,y position** in order for the Prakash NFT Auto-Compose Engine to properly compose and layer the resultant image/NFT:

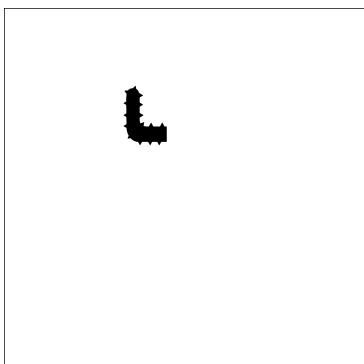
1. Body (trait #1; color = cyan)



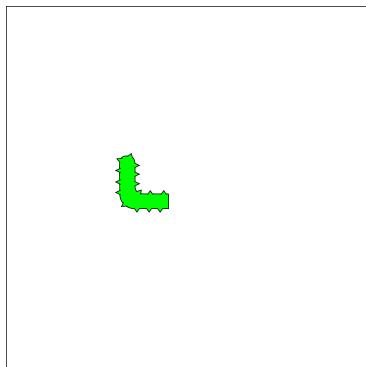
2. Pot (trait #2; color = dark brown)



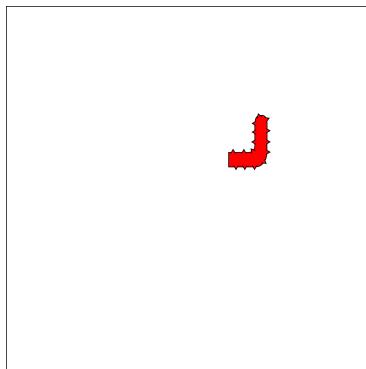
3. Left Upper Arm (trait #3; color = black)



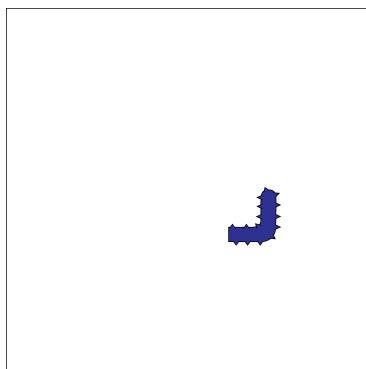
4. Left Lower Arm (trait #4; color = green)



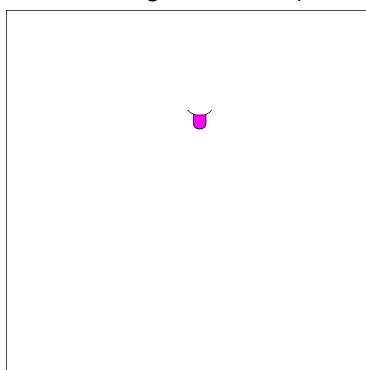
5. Right Upper Arm (trait #5; color = red)



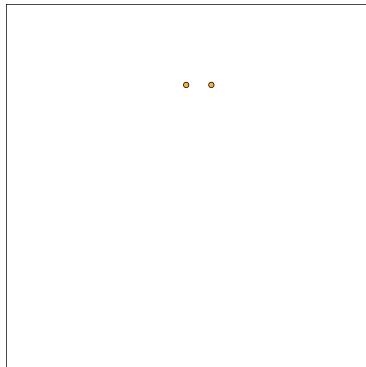
6. Right Lower Arm (trait #6; color = purple)



7. Tongue/Mouth (trait #7; color = pink)



8. Pupils/Eyes (trait #8; color = orange)



1. Body (all assets/all variations - just color variations in this example series):



2. Pot (all assets/all variations - just color variations in this example series):



3. And so on for all the traits for the “Cactus Club”

To Be Continued... NFT Bears Case Study.