Updated:
December 27th 2021. These notes cover up until the release of Alpha 1 on December 11th 2021

Personal

- I was commenting earlier how I work 2X as much as my last real job, but this is 2X as fun.
- Basically when I quit my job I decided it was forever that I would be on my own because I won't be able to explain the hole in my resume. I was tired of corporate life anyway. No looking back now, and in retrospect I made the right decision. The project was much smaller at the time so it was a bit of a leap of faith!
- Alpha will be out Soon™ and 2.X will be done WhenItsDone™ 🙂
- 2.X alpha will be out soon. 2.X won't be fully complete for years. Releases will happen as we go. Just like 1.X. It was never really done either, we just moved to a reset with 2.X and keep going.

Q- are you Chris Roberts ?
A-If I was Chris Roberts 1.X would still not be out yet and I would have presold you 30 different models by now. 🙂

Q- If the future is in the metaverse, I don't want the Facebook's metaverse, I want the @meshedvr metaverse 😆
A-I think the 2.X metaverse is not going to support more than 3 or 4 people in a room. Not very meta. How about a miniverse? 🙂

A- I do listen to music in the cave. I have wide musical tastes. Currently listening to Rush Permanent Waves.(*Editor's note: Fucking great album*) I often put on movie soundtracks because music with lyrics can be too distracting if I'm really deep into something.

A-My personal favorite shameless knockoff is Virt-U-Mate. lol. At least the top review mentions this is not Virta-a-Mate, so free advertising for us.

Q- Can you do a copyright claim to get it removed?
A- Not worth my time or money. Too much work for no benefit.

Alpha 1

Q-Do we have visibility on an approximate exit time, 1 or 2 months or 6 months or 1 year?
A-I'm not sure what you mean on exit time, but I'm no longer giving out estimated dates for releases or when specific features will be completed. I'm terrible at estimating and there are always unforeseen issues that require time consuming experimentation and workarounds.

Q-so for the first alpha that is going out :soon: *will it only have 1 female model or will there be male models as well? And can we add more than one in a scene?*
A- The 1st alpha will only have a single female model and you cannot add additional models or objects.

Q-Is it possible Instead of doing an alpha build every month or 2 months. Doing smaller updates like every 2 weeks?

A- Probably not going to release every 1 to 2 weeks. It is hard to get anything really tangible done in that time frame. I'm not making any promises on when releases will come out. They will be released when it makes sense for the features being worked on. Some things take longer than others.

Q-what are you currently working on / what is missing to the point where you want to get with the alpha?
A- I'm working on materials and shaders to complete what I started there. I opened a can-of-worms and I need to get closure on that before release. After that, I would like to see if I can get basic VR support in, but that might not make it in. I had it working in an earlier 2.X project to where you could at least view the scene. That is basically it for the alpha, and then it will be released after some testing.

A-Yeah 2.X will be a fun side distraction but you won't be using it for very long. There are no saves and you are limited by the included models and all the stuff in the UI. Think of this 1st alpha as a tech demo and glimpse into the future.

A-Things on 2.X are going a lot faster than the original 1.X development. That is what experience does, and I have a 1.X codebase and setup I can pull many things from. Many things from 1.X can be ported/adapted to 2.X without writing from scratch. I'll just say this. 2.X will take years to get to where 1.X is now. I was developing 1.X part time for years before the Patreon project even launched. The oldest stuff in 1.X is probably 5 years old now, and there were quite a few experiments before that.

A-Also note at some point I will disappear deep into the cave again and be less responsive here *(Discord 2X channel*). Once the alpha is out there will probably be a lot of questions and suggestions, and I'll stick around for that. Then back into the cave to get alpha.2 going (start on skin physics, parameter and save/load system, real UI).

A-1st release will be 2.alpha.1. Then 2.alpha.2, etc. until I feel it could be a proper beta or just regular release

Q-What is the goal of the first alpha version?
A-To demonstrate some of the new technology pieces that will make up 2.X

Q-When can we expect version 2.0?
A-Soon™ + 1 hour each time someone asks that question. 🤣

Q-A big part of what made the 1.x learning curve so steep was the lack of documentation. Will 2.x have official documentation from the start? With updates coming with each new release?
A-Yeah step 1 is making it more intuitive and less overwhelming with too many options thrown at the user at the start. 2.X will have layered complexity, meaning ease of use for basic stuff while still allow more advanced usage for those that seek it. The UX itself will be customizable as well for someone who has a specialty use and wants to improve these ease-of-use for specific things. The upcoming alpha already has a Help button with some info on how to use the alpha, so I guess that is a good start. We will try to do better with tutorials and docs for 2.X, but I doubt it will be perfect.

A-The upcoming alpha is only including a single model: Genesis 8 female base + Genesis 8 female genitalia for demo purposes only.

Q-Will VAM2 be available at Steam store?
A-We will hopefully take 2.X to a retail market, but that is a long way in the future (years). I do think we could launch on Steam based on what else is in there.

A-SuperSoon™

A- From 1.2 release is pretty close!

A-Or the old banner from around that same time (2017):

A- https://hub.virtamate.com/threads/2-x-tech-alpha-1-is-released.14928/

Plugins

Q-What is your view on the integration / use of ML tech like GPT-3 to VAM? This could help build an AI capable of directly talking to the player.
A-Everything in 2.X is being created in a way to allow external interface in both directions. This means a 3rd party program could be written to "see" what a person atom is seeing, process the data, and send 2.X something back in reaction. We are not going to build in this kind of stuff, but instead use the plugin or external software approach. This gives maximum flexibility. I absolutely think having AI plugged into 2.X is the future of virtual adult experiences.

Q-Can't wait to hear more about the communication with external processes...used to do a lot of stuff with Unity and OSC...are you all planning on your own protocol??
A- I'm not sure yet. I know 2.X will have an official API and I plan on making that API callable from external software. Part of the API will include rendering a texture from a custom camera, so you could do some image-based processing. That camera could render with custom lighting to make it easier for AI processing.

Q-plugins will still kinda have the ability to communicate with unity objects/classes like usual?
A-Not 100% sure on that just yet, but probably. One thing that was a huge pain in 1.X was security. It is easier to provide a safe API than it is to try to find all the loopholes that c# and the Unity libraries allow. The official API also makes it easier to make new releases without breaking existing plugins. A lot of times in 1.X plugins used interfaces I considered internal and when I changed them, the plugin would break. I'm hoping to avoid that. Plugins are a ways off in 2.X. I'll revisit when it gets closer. I'm hoping to involve plugin creators in the API development itself...

Q-Have you looked at Direct'2'Storage yet? Don't know if they've released anything in the way of an API for it yet.
A-I think a customized cache system like what was built in 1.X is going to beat out DirectStorage.

Q-will 2x let us reorganize plugins and addon packages, like with a tiered favourite system?
A- I haven't given this part much thought yet. I will consider improvements when it gets to that point.

Q-What I'm really looking forward to is understanding what the extensibility will look like

A-At this stage in development I only have ideas of what I want to do with the plugin architecture. I would like to have an official API that plugins use and avoid other internal interfaces that might change during development. In fact the plan is some of the functionality we are going to deliver is going to be built using this plugin API. We will probably even release the source for much of it so it can be improved upon or modified. What will be closed is some of the key technology pieces that we develop, like the skin physics system. The UI/UX is also going to be modifiable/extensible. We also want to add an external interface so 2.X can talk to other programs and vice versa. The thought here is more complex stuff that can't easily be done with a plugin, like integrating with custom hardware or interfacing with some other software that requires 3rd party libraries. I'm sure I will be chatting with you a lot more as we get to this part of the development.🙂

A-I don't think anything in the core or API is going to limit animation. The animation system itself might be limited, but it will be because it isn't the primary focus. Plugins will be allowed to have full scale UI/UX in 2.X. You could even have a second program running outside of VaM controlling what happens inside of it. So you could end up with a multi-window setup. And we will have some way to pass that external program a render texture that can display a view from the in-game camera. So I expect we'll end up with some really interesting stuff in 2.X once that is all ready. The built-in animation system is going to focus on simplicity and ease-of-use. We can revisit all of this when we get closer to that.

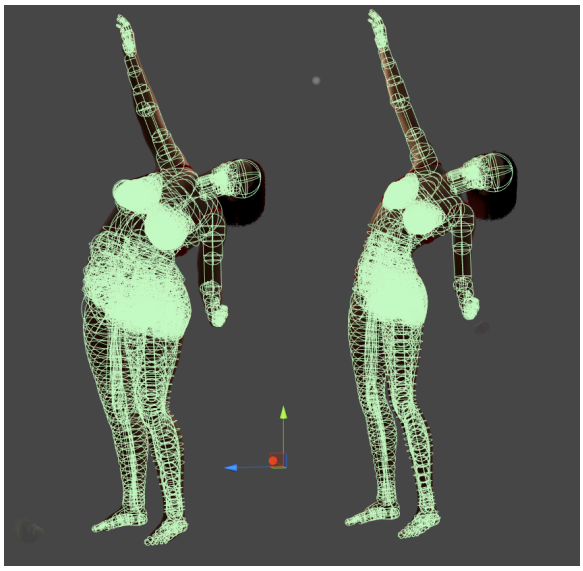
A-Plugins might actually be built in Unity and imported into VaM in that manner. That will allow making custom UI using the Unity UI tools. The 2.X core will be simpler, focused on the systems like control, physics, rendering, and a baseline parameter/save/load system. Everything else will be built on top of that and will go through the API to interact with that core system.

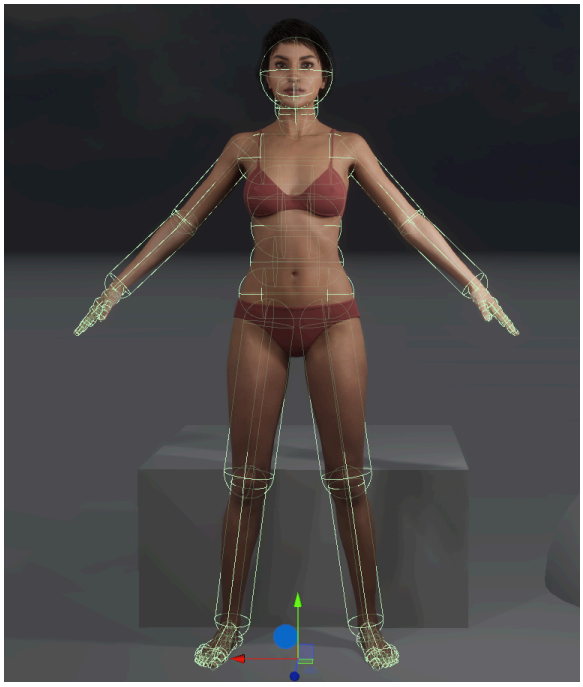
A-I don't have any details on the 2.X API yet. It will be worked on SomewhatSoon™ as it will be needed for some of the automation systems we want to add (eye gaze, etc.) I'm keeping external programs in mind for 2.X while developing API. I would like it to be possible to fully control and get all data from VaM from another program.

Physics

Q-In what state is the physics system in? Any chance we can get a demonstration of that soon?
A- The joint physics is what we have been working on. That was shown in the last sneak peek video. It has been further tuned since then. The 1st alpha will have a very primitive collision. The soft-body and skin-accurate collision system will likely be worked on immediately after the 1st alpha is out.

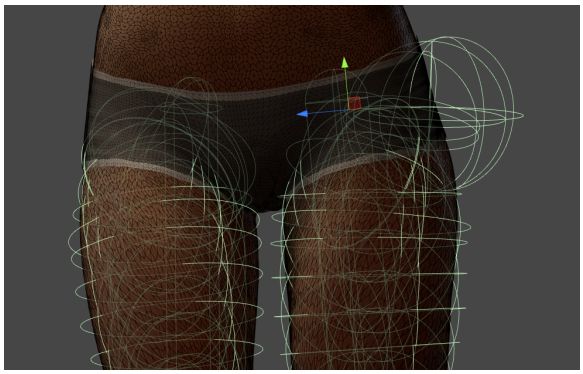
Q-What do you mean when you say skin physics?
A- The goal for 2.X is to have nearly whole-body soft physics, and not just the 1.X system covering more area. The goal is to support a fat texture map that can have varying amounts of softness over the skin. This will be very hard to pull off without killing performance. We may have to do this on the GPU which would make it a lot more complex to feed back to the regular physics system which handles joint physics. The alpha demo that is coming out has fixed colliders that are not skin accurate and don't change with morphs. The whole body skin physics will work with any shape. 1.X had a combination of auto colliders and soft-body colliders. Soft-body only covered breasts, butt, female genitalia, and lips. Auto-colliders were another tech we added that allowed many colliders per joint that adapted somewhat to morphs, but they had limits and could become quite inaccurate depending on the morphs. Here is picture of the 1.X collider system (both auto colliders and soft-body) and Current 2.X alpha colliders:





Q-Something like a height- or weight map for soft body physics?
A-Yeah a bit. You will have a texture map where black means hard, and white means soft. Hard areas of the mesh won't be able to be pressed or jiggle. Soft areas will have the most amount of press and jiggle. There may need to be multiple layers for really fatty areas to get proper jiggle.

Q-It sounds like the solution you're heading towards is something out of a white paper because I have no idea how you'd work to accomplish that. The current soft bodies in vam 1.x behave kind of like collision enabled jiggle bones more than a truly soft body but I might be wrong about that.
A-Yeah this picture here shows how big the 1.X auto colliders are. This one showed an issue that was fixed but demonstrates the size and why they would not work with all morphs. For 2.X there will be a collider per base vertex, or nearly 17000 colliders. 1.X has around 1000 colliders, and most of those were in breast and butt area.



Q- How will that not bring performance down?
A-The magic of Unity's new c# jobs system, and if that isn't enough, a GPU simulation. GPU can handle it easily, but there is a latency issue getting data back on the CPU to allow soft-body to push back on other colliders and joints. I'm going to keep soft-body out of Physx. Physx can't handle that many colliders. I'm

writing a custom collision system. Preliminary tests look promising for performance in c# jobs / burst in Unity. It can really handle a ton of data and use all your CPU cores.The new skinning system uses it and it is impressive.

Q- This sounds promising, let's just hope there is still enough room for the other things eating cpu and gpu as well like hair.
A-Yeah skin physics is the only big question mark remaining. It is not clear if what I want will be possible, but I'm pretty hopeful just based on some data I collected using the system. And I do know GPU can handle it easily (cloth physics is actually more complex and with 100000s of colliders).

Q- how bad do you think it would be to have a frame or two of latency on the skinned physics interacting with regular colliders?
A-1.X has some issues with multiple systems running async and not aligned with physics. I had to add in a prediction system which can cause lag between skin and clothing. 2.X is being built fully synchronous. But we could do skin physics with 1 frame lag to allow more time for processing. To do this we will need to make sure everything else is aligned to account for the latency to prevent the issues you see in 1.X. 1.X had asynchronous skinning as well that ran on render cycle, not the physics cycle. This could cause a big lag and disconnect between skin and what physics engine needed. Skinning on 2.X will run on physics update (probably 50fps fixed) and rendering will run at desired display rate (variable for VR) and use interpolation for display. This combined with much higher number of skin colliders should give 100% collision accuracy. Also - in 2.X you will be able to pin other objects to skin vertices and they will move and render in sync. This will allow things like jewelry, hats, etc. The collision system will use partitioning for collision so it won't be 100% brute force. I'm hoping to improve the cloth and hair system with partitioning as well which will allow for better performance and the possibility of self-collision.The fastest collisions to check are sphere to sphere. I'm envisioning a multi-pass sphere partitioning system that gets smaller and smaller with each pass. I have lots of ideas based on 1.X experience. The problem I'm going to have is skin physics working with mesh colliders in Physx. I don't have a great solution for that. I can handle the other primitives pretty easily. 2.X may just not support mesh colliders. They are quite expensive anyway.

Q-Would be nice to see if that type of thing can work with hair (hats/hair jewelry etc) with proper collisions
A-Attachments to hair would be cool, but that is going to take a system that can take data from the GPU and get it back to the CPU to drive the position of the attachment. This particular thing of going from GPU->CPU is fraught with peril and is to generally be avoided. It can take several frames of lag to get the data back. This has improved somewhat in recent years, but it could be tough.

Q-What does 1.x even use mesh colliders for right now? It appears bodies use capsules unless I'm mistaken. Are props using mesh colliders?
A-Yeah they are used for nearly all environments. In some cases we took the time to convert mesh colliders to primitive colliders to speed things up, but it can be a huge pain. There are a few tools out there that can help automate the conversion. I want 2.X to be able to support custom atoms that people might make using assets they have, and that means they will have to take the time to define primitive colliders.

Q-Did you look into bullet physics..skyrim modder use it for soft body physics..now they have ported it to cuda,..and can use desired gpu for calculation
A- I can look at it. I also looked at Unity's new physics and Havok physics systems, but they are too premature and were missing key features, like joint drive. I think the fastest solution for skin physics is going to be custom because I can strip away everything that is not needed that weighs down a traditional

physics system. I also want a more unified physics approach for cloth and hair. I'm thinking skin, cloth, and hair can all be running on a similar system.

Q-How will the new skinning and physics play with the hands (fingers being like rubber) and something like a boob job?
A- You'll be able to check out improved fingers in the upcoming alpha. They are much less rubbery. Collision accuracy is also increasing (not yet in the alpha) and will make interaction with hands more natural.

Q-When the VR version of VAM 2.0 will be available, can we get a feature to grab and drag geometry itself not the control nodes?
A-In the alpha you can grab the model anywhere, but it is not 100% skin accurate yet. That won't happen until we have skin physics working.

Q-With the new systems you have in the pipeline, will the physics play nicer with body scaling? Like in 1.x the control nodes won't keep alignment when scaling.
A-The new skin collision system we have planned should scale a lot better. There is weirdness with Physx and scaling at extremes so I can't promise anything too far out of range will work.

Q-skin physics is still going to come after the first preview alpha, or did you already do a bit of work on it?
A-Skin physics will come after 1st preview alpha. I know "how" I want it to work though. We need to do experiments to see how to do it with best possible performance. I'm not 100% sure we can do it on CPU with the performance needed so it might be a trickier CPU/GPU combination.

Models and Morphs

Q-Will morph import and morphs sliders be available in the 2nd alpha ? Or is it something very far away?
A-Morphs are working and there are a few you can play with in the 1st alpha. The 1st alpha has a throwaway UI and the parameter system is not yet done. I think the work that will happen after 1st alpha is skin physics, parameter and save/load system, and start of official UI. That will allow more morphs to be added to interface. Also - bone movement morphs are not yet functional. I plan to tackle that for 2nd alpha as well. Morph import is a ways off as I need to provide a solution that is easy to use. The current solution only works for built-in morphs.

Q-If we are getting some built in morphs for first alpha can I request at least a nose category like we have in 1x like nose tip round, bottom square, septum width or g8 equivalents, similarly eye, mouth, cheeks, chin and jaw categories?
A-1st alpha will only have like 3 morphs to play with. There is not room on the UI to put more. Until I build the real parameter and UI system I don't want to pile more in. The 1st alpha is more of a tech demo...I expect the 2nd alpha to have more extensive morph system, including bone movement morphs which are currently not functional.

Q-What about the female genital in v2.0 ? Is this going to be similar to the New Genital for G8 or something?
A-We are evaluating several genitals. The alpha includes the standard gens, but it is not yet functional. We have run into a some issues with some of the 3rd part gens that we are trying to work out as the standard gens has poor UV texture coverage and the mesh isn't that great.

Q-I just want really good tech to support enormous boobage.
A- Custom models will be supported so if the out-of-the-box models can't go extreme enough, a custom model could be added. But I think the skin physics system will be flexible enough to allow it. The biggest issue you would face with built-in models is enormous boobs stretch that area of the mesh to the degree that the skin collision system might not work very well as there are not enough mesh vertices to cover the area. And the textures also get really stretched so without a custom UV and texture set it won't look very good. Both of those are solved by a custom model. Maybe some kind of giant boobs graft would work for this. We will be providing the tools we use to make the models, so someone could make a custom model using custom grafts and achieve this.

Q-smoothing will work better with quad topology....
A-For now 2.X is not using smoothing on character models, but may add it later. Instead we are favoring HD mesh/skin with no smoothing. This might be an issue in some poses so we will consider a smoothing system later on. It won't be as aggressive as the 1.X smoothing system was which tried to correct skinning errors with pretty high settings. Subdivision is not necessary and is pretty expensive to compute at high frame rates. For the character models, we are instead doing the skinning on the already subdivided mesh. We are importing the HD mesh and morphs. We have subdivision level 1 running in the alpha. subdivision 2 could also work, but it is a bit too expensive to run on the CPU, so it will be run on the GPU instead for final rendering. The base mesh skin will be run on the CPU for physics. A fast smoothing algorithm might be needed to achieve the best results since we are not using subd and the smoothing that is inherent in that process.

Q-Regarding importing custom models.
A-You will be able to add custom models to 2.X. You can use custom grafts. You will have to use the same tools we are using to build the current person atoms to do it so it will be more involved. We are separating the content creation part from the engine to avoid having the engine be overly complex and stuck with subpar solutions for problems that are more easily solved by dedicated external tools.

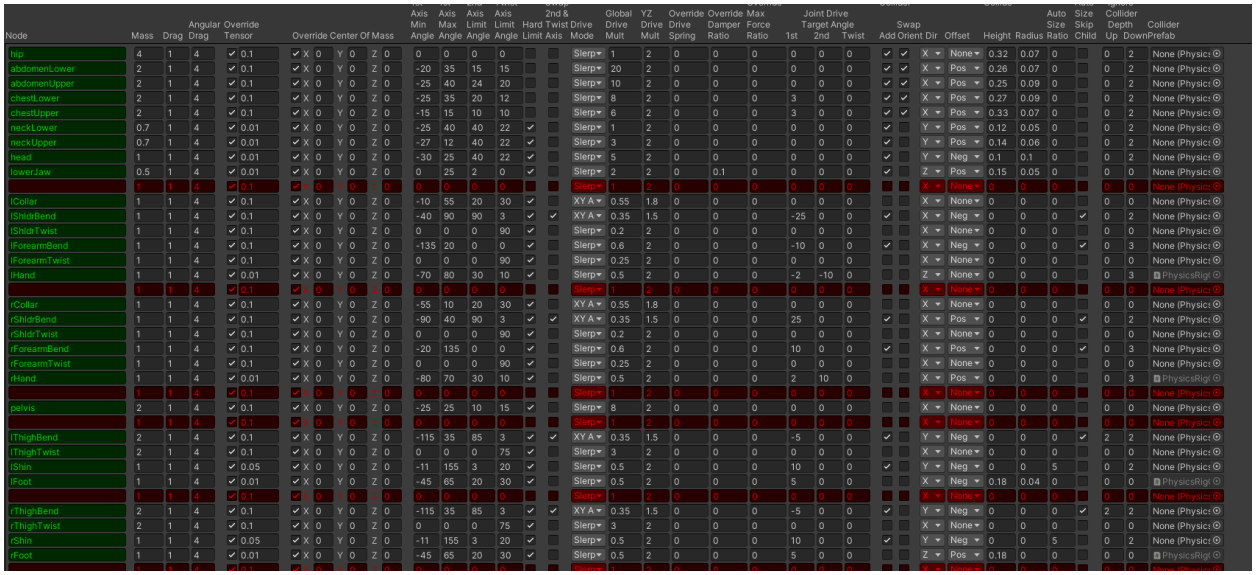
Q-will we be needing any expensive software to use the tools?
A-We are providing Unity tools for the custom atom creation process. You could use Daz Studio and export your custom model from there to our Unity tools that allow it to work like the included models work. If you want to do things like creating textures you might want a tool like Substance Painter, but this is no different than how it was on 1.X. You should be able to do just about everything with free tools (including free version of Unity) if you know what you are doing.

Q-Can you buy all kind of things in dazstore and then import it in vam2x without any problem?
A- It isn't going to be as easy to import stuff like it was in 1.X, although we will likely provide an easy clothing import tool. DAZ Studio rigged hair never really worked, but you could import it as a custom atom in 2.X. This will be more difficult than creating hair using the built-in hair sim like exists in 1.X. Not everything is just going to work perfectly without some manual work. In short, you can do more in 2.X than you could in 1.X, but the import process will be more complex. Once created, the item will be easier to use in 2.X than it was in 1.X.

Q-That's okay we will make sure the hair is superior to DAZ by then anyways
A-The current HDRP hair shader seems really expensive. You'll see this for yourself if you zoom in on head and watch your FPS drop. I have a toggle so you can turn it off. I do think with simulated hair and a proper HDRP shader that is geared for strands is going to look really good.

Q-when you state we'll be able to "add custom models", that aren't G8, or maybe aren't even Daz models at all, are you saying we'll be able to bring in our own topology, UV's, textures etc., from the ground up and hook it up to VaM's skinning and physics engine with the unity tools, and it will work? Will there be topological limitations (vertex counts, etc.) that models will need to conform to, and what about rigging - might it be possible to import a custom rig that would work with VaM's vertex skinning/collision system as well?

A-Yeah "add custom models" is just what you described. For the mesh you will likely want multiple detail levels. We are about to start on the skin physics system so I don't have exact details, but no more than 20000 vertices for the lowest res mesh as that is what G8 is. You can have higher resolution mesh (HD) for final rendering. You will be able to have own topology, UVS, textures, and bones driving the skin. You will be able to hook it up to everything VaM can do natively. We worked on a system that will make it a lot easier to do this. The physics rigging is quite complex and we created a table-style tool to make it quicker. You can use G8 as a base or guide for your own model to know what values to put in. Here is a screenshot of that system:



A-You will need to create and import any morphs you want to work with this custom model. Built-in morphs will only work on exact combination of base model + grafts that we have chosen. In 2.X we don't do geo-grafting internally. We instead import the combination model + morphs. This is a pretty big performance gain and gives better results than the in-game realtime grafting we had in 1.X. That system would not really be capable of supporting custom models.

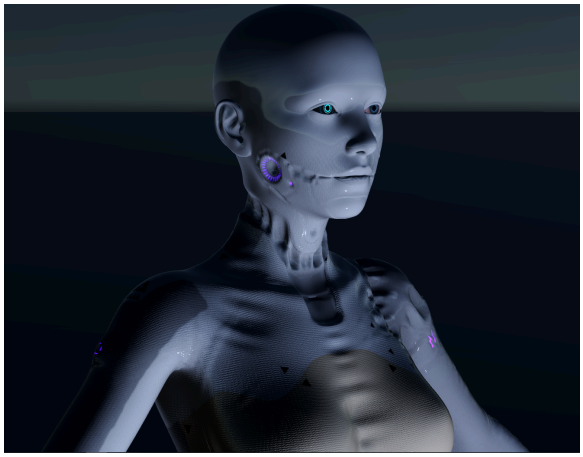
Q-2.X allows to include Vam 1.20 material? Like Assetbundles, saves and morphs?

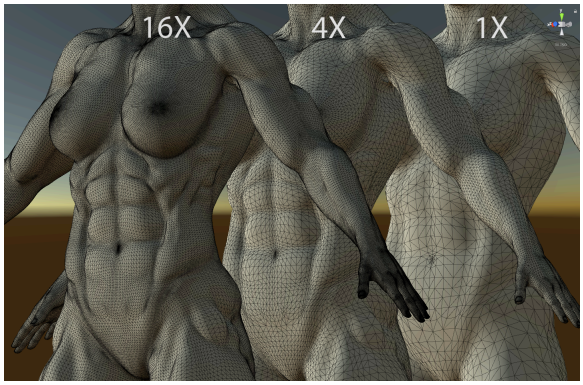
A-2.X won't be able to load any 1.X content. Some 1.X content (like clothing and previous Unity custom assets) could be ported over to work.

Q-IDK if this has been asked but is it possible to port and have these characters from DAZ into 2.X?
https://www.daz3d.com/mia-lvl-4-hd-morph-for-genesis-8-females

A-That is level 4 subdivision. I don't think that will be possible real-time as it over 1 million vertices and polygons. 1.X used base mesh + tessellation in some areas to get smoother curves. The upcoming alpha is using subdivision level 1 running on the CPU. Level 2 (any maybe level 3) will be possible once we have the skinning engine running on the GPU. If you run level 2 model on CPU now it takes up about 1/2 the CPU budget so that might give an idea of how much it costs. Each additional level is 4x the number of

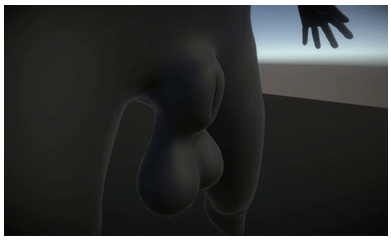
polygons from the previous level. So if base is 20K polys, level 1 is 80K, level 2 is 320K, level 3 is 1.28M, level 4 is 5.12M. I did post this a while back on the 2.X May update on the Hub. Here is level 2 (16X) mesh.





A-I think Level 2 (16X) is the sweet spot. At level 3 I don't think it will be possible to run the morphing engine + skinning real-time. If the figure was pre-morphed it might work. Level 4 is really offline rendering only. I'm not even sure it could be imported in to look at in a static manner. Yeah actually the highest it looks like you could export out of Daz Studio is level 3. That morph set probably works at lower subdivision levels though. They just like to advertise level 4 as it implies very fine details.

Q-thought this was pretty cool. Wondering if we can get similar on male penis in vam 2.xx, foreskin wise.



A-I wonder how much of that is physics that can adapt to any situation vs a hardwired rig using a combination of joints and mesh morphs. I hope to achieve what you are seeing with orifices using the soft body system.

Q-For male characters, will we finally have separate controllers for each ball?
A-We are looking at several male genital grafts, and I believe they all have separate joints for each ball, so yes. There will also be soft-body physics on top of that for additional dynamic squish and stretch.

Q-Regarding the process of importing morphs and models into V2.
A-I have actually already worked on Unity tools to make the custom model import 100x easier than it otherwise would be. Once it is more streamlined, automated, and stable, we will release to community along with a how-to video. One of the top goals of 2.X is improved usability, and that includes documentation, tutorials, etc. as well as improved interface. The import pipeline will be more complex than 1.X, but it also comes with a lot more capability (custom grafts or even non-daz3d models).

Q-my biggest fear was having a long process to get models in.
A-Unfortunately the custom morph import is the hardest one. Due to how HD meshes with grafts work, morphs must be exported out of DAZ Studio on the complete model (base mesh + grafts), and only apply

to that specific combination. 1.X has a basic real-time grafting engine built in but it did not match DAZ Studio exactly so cannot be used here as it won't give good results for all grafts. That real-time grafting also cost performance. So 2.X does not have built-in grafting. Instead it relies on the model coming in complete with morphs and skinning specific to that model. I am likely to make a standalone program that can take the morphed mesh (obj or fbx format) and convert to a morph for that model. You will have to provide morphed mesh in base resolution and various subdivision levels (probably both 1 and 2).

Q-So technically you could load an entire appearance in 2.x through an .fbx file instead of a morph altering the default appearance?
A-The 2.X runtime engine is not going to be able to load fbx. You will have to use tools we provide (for Unity) + Unity engine fbx import and convert that to a binary morph file + meta file that applies to a specific model. If we build a standalone tool, it will do the same but not require use of Unity. This would just be to make the process easier. But essentially making a morph is taking the source mesh and comparing it to the morphed mesh and making vertex deltas file that gets saved in format that 2.X can read. Depending on how many vertices are moved in the morph, the delta files for the higher subdivision meshes could get somewhat sizeable. At a minimum it is 12 bytes per morphed vertex. If you morphed entire subdivision 2 mesh that would be a bit over 3 MB per morph.

Q-Will we still have sliders so we can create stuff in 2.x or will everything have to be done outside and imported in through this process?
A-You will be able to modify everything in 2.X using sliders. Once morphs are imported you can use sliders to apply them. We will support runtime texture conversion and loading, but if you want optimal loading performance, you will want to utilized a new way to make texture packs (using Unity) that work in 2.X. The biggest difference is 2.X won't have all the conversion tools. You won't be able to read duf/dsf files direct into VaM and have them convert to morphs or clothing items directly. Instead you will need to use external tools for that.

Q-one of the things I love about 1.x is playing with the morph sliders in game.
A-Yeah I agree that is important and fun! For those seeking the best results, they may want to create a custom person atom with custom grafts and morphs pre-applied along with custom JCMs specific for that model. Then on top of that use custom normal maps to get the last level of details that the subdivision level 2 mesh itself could not provide. Daz3d has actually done this for a lot of their self-produced models. For example the Kala 8 model comes with some JCMs that only should be applied if the Kala 8 body morph is applied. I don't see a really easy way to support this without using custom-model approach. I don't want to have 1000s of JCMs that are only partially or fully applied depending on how the morphs are set. It might be all or nothing enable of specific set of JCMs. Also of note is those wanting to make custom morphs for the built-in models are going to have to purchase the same genital grafts we do in order to make morphs. This is due to how the 2.X morphs are on the combined mesh and not the base mesh or graft meshes separately. We may provide multiple female, male, and futa models with different genital grafts. The Daz3d genital grafts are ok, but provide poor UV coverage. We have been extensively evaluating 3rd party genital grafts.

Q-So gone are the days of being able to mix and match morphs? Each morph is technically going to be tailored to each model/graft variation, so that means a specialized model can only use the set of morphs that the creator includes?
A-Yeah because the 2.X engine will not have real-time grafting. The 1.X real-time grafting engine barely worked using standard gens and in many cases (depending on morphs applied) looked bad and did not match what was seen in DAZ Studio. The grafting system in 1.X was also pretty costly, and now with HD

mesh support would not really be possible without causing major performance issues. So morph files that load in 2.X are really morphs for the entire model combination (base mesh + chosen graft meshes).If someone creates a specialized model (not using 3rd party copyrighted grafts), they might provide the source mesh in fbx/obj format so others could make morphs. But the output morphs are only for that model. Or they could provide source dsf for their grafts and then you could use DAZ Studio to make new combinations and export that to 2.X. It does kind of suck, but this is the only way to really support HD mesh and morphs. An example of where 1.X grafting fails. Dial in one of the fat morphs and compare to what you see in DAZ Studio. It is quite different. The male gens get distorted quite easily. People made correction morphs to fix these issues. DAZ Studio doesn't provide info on how their grafting system actually works, so 1.X grafting is not 1-to-1 match. It took a lot of experimentation to even get it to where it is now - just good enough.

Q-How are expression morphs and such going to work then? Do they really have to be done for each model? are they provided for the standard models?
A-2.X has realtime morph engine. Realtime grafting is what is being dropped. Morphs must be imported with model as a whole (base + grafts). That doesn't limit what morphs can be used, it just makes it a bit harder to import them.The included models will include as many morphs as we can pack in. We can add more if requested. Expression morphs will be included. G8 actually uses a face rig (bones) for many of the morphs. What I see is combination of bone movement + mesh changes. This will all work correctly in 2.X. This image shows the upper face rig "bones"

These are used in expressions, but these could also be used to morph the face without using morphs. I'm not sure how we will extend this to the user interface yet.

Q-to clarify, morphs that comes with one model, (boobs-Bigger) will not work with another unless they are same graft and subD level right? because it needs to be the same geometry?
A-Yeah morphs are on the combined geometry, meaning base mesh + graft meshes at multiple sub-division levels. It looks like for 2.X you will need to provide base level and subdivision levels 1 (4X) and 2 (16X). The physics system will use the base mesh, and the rendering system can use base, subd level 1, or subd level 2 depending on user preferences and other specific setting for the model (you might want to lower-cost background model running at base level). Also - since Daz3d has HD morphs only

available to specific PAs, we will provide a way to create HD morphs directly from an imported morphed mesh. You could export subd level 2 mesh to a sculpting program, sculpt the HD morph, and then import the morphed HD mesh back into our system and it will create the HD morph. It will just calculate per-vertex changes.

Q- if I'm understanding how custom morphs will work, it's going to make character creation much less accessible to the average user.
A-Morph creation will be less accessible than it was on 1.X. In some ways this might be a good thing. 1.X was littered with morphs and it was a big mess. 2.X morphs will be much more organized, including duplicate removal and removal of morphs that don't fit the character they are trying to be applied to.

Q- any thoughts or ideas on how to get a VAM model exported into VRChat?
A-We don't allow export since the built-in assets are copyrighted. The best you can export is a static mesh in .obj format.

Performance

Q-How is the performance compared to 1x and what about shadow quality?
A-Current performance won't say much about the final performance because skin physics is not yet implemented, and it will be the biggest performance hog. HDRP rendering from Unity has a higher fixed cost than the rendering system used in 1.X, but it looks so much better and supports more lights without much additional cost per light. Shadows in 1.X used a special 3rd party soft-shadow solution, and 2.X uses HDRP shadows which tried to so some of what that 1.X soft shadow system did. I think shadows are overall improved, but they are quite expensive at the highest settings. Raytraced shadows are also a possibility which will look the best, but are going to require a really good GPU to work at the framerates needed for VR.

Q-will it be possible in V2 to have a large group of people as background objects?
A-It should be possible. We are planning on support LOD (level of detail) which will be directly controllable. 2.X will also support custom atoms so you could make your own crowd using low-cost assets if the LOD system is not enough.

Q-There are such things as hardware texture compression, such as DXT1 format for example which is used by Unity engine. Its not as space efficient as JPG on disk, but it can stay compressed in graphics memory as it can be decompressed at runtime by the GPU. That part should be the same as VaM's cache system. The difference to DirectStorage is that it can bypass CPU/RAM and go to VRAM directly. So, yeah....VaMs current system can't beat it, but may be it can be modified to use it.
A-Ah indeed I didn't understand what it does. One thing I'm addressing is while 2.X will support texture loading and conversion, the preferred method is going to be to create a texture pack in Unity and have that loaded through assetbundle. Textures in Unity assetbundles are already processed and ready for direct GPU load. If you notice the built-in skins load very quickly for that reason. Runtime loaded textures are not and require processing (convert to proper format, compression, create mip maps, load from CPU to GPU memory). One thing I did with the texture cache is convert the textures to GPU format including compression and mips so they can be more quickly loaded. It still is slower than loading textures from assetbundles. I assume Unity has some faster internal methods that bypass the Texture2D class. Texture

loading is probably the slowest thing in the loading process now, but large animations can also be slow. We plan to have some binary format support for large data sets like that to also improve loading.

Q- vam uses cpu time to process texture, this was main reason of high loading time even in nvme ssd. Direct storage should help there but its going to too much storage i fear
A-The vam texture caching system fixed the processing issue. It is basically just uses Texture2D LoadRawTextureData and Apply functions. But yeah I think Unity has some way to bypass those and load direct to GPU if the textures are in assetbundles.

Q- Will 2.x have improved performance?
A- 2.X should have improved performance per person, but if you dial up all the quality settings you are going to have worse performance than 1.X. The goal is for 2.X to have scalable performance options. HDRP has a higher baseline CPU cost than the rendering system in 1.X. 2.X should have lower physics CPU cost. It might balance out. Everything is a tradeoff of quality vs performance.

Q- I imagine by the time it's all done there will be several fps hogs and users may have to calibrate settings depending on what they want to achieve.
A-Yeah - I'm already seeing FPS could be a challenge for higher resolutions like 4K monitors and VR. Fortunately DLSS exists now. 🙂There are zillions of features that can be turned on/off. Some are real performance hogs. The preferences page is going to be a bit ridiculous. I hope I can get it down to a few fixed Low, Medium, High, Ultra settings that generally work for most people.

Q- Any advice on an affordable future PC for Vam2?
A- It is a bit early to give strong recommendations on hardware as we have not implemented the skin physics system which will probably be the most demanding feature in 2.X. Based on what I see so far the new rendering system (HDRP in Unity) is pretty demanding of both CPU and GPU. But Unity is continuing to make this better over time and there are tons of options that can be turned off to improve performance. If you want to see 2.X in its highest visual glory, you are going to want the best GPU you can get. For the skinning system I created for 2.X, it utilizes Unity's new C# jobs system which can run jobs on every CPU core you have. But it does this in bursts and the skinning system is not very demanding, so while it uses all cores, you won't see all cores at 100% all the time. I also plan to use this jobs system for the skin physics if possible. So having a high-end, high-core count CPU may also be desirable for 2.X. I want to make 2.X as scalable to various hardware as possible, better than what 1.X has done. I also want 2.X to manage memory better than 1.X. 16GB is probably a good target. The first alpha release will be out Soon™ so you can get some early performance tests from that, but that alpha is also not very optimized and has almost all options set to High or better, so it might not be the best measure. I was able to run it in VR with my 4 year old i7-7820X & 1080Ti system without major issues. For desktop 1070 should be fine.

Features

Q-What are the plans for fluid physics?
A-Yes 2.X will have fluid physics. The reason we didn't do fluid physics in 1.X is we did not think the skin collision accuracy was high enough to do it well. 2.X should have much higher accuracy.

Q-Anything about handtracking?

A-I'll be adding support for the latest Ultraleap drivers, and I'll also try to get in Quest 2 and other headsets that have added handtracking. I really hope a standard develops so I only need to code up a single system, but that seems unlikely. In 1.X I created my own mapping system so I could isolate the input tracking from the output finger drive.

Q-Vam needs frames and realtime posing to avoid vr sickness and let us animate hella fast
A- 2.X should be able to do both...

Q- id love for 2.0 is easier way to manage mods.
A- Yeah duplicates will be pruned by getting hash on all data files and hiding any duplication of content, but the var files themselves could still contain duplicates. We are not going to move to individual file publish system. There is still a challenge of knowing which var contains the master copy since when you use an item and then make your own var, it would be good to use the proper original var file reference and not a duplicate. Hopefully duplication will be less of a problem. We will add whatever we can to help minimize this issue.

Q-You once said that new parts built at the end in vam 1 could be used in vam 2 to speed up vam2. Did it work out in the end or did things change so quickly you have to rethink/rewrite all that stuff?
A- Yeah some stuff from 1.X will be able to be carried over pretty easily. A lot of the file and security system work that was done near the end to support var packages will carry over. A lot of the parameter system will feed into 2.X, although I have improvements planned for animation and undo/redo system. Lot's of the work from 1.X makes 2.X way easier to make. I'm going to improve everything possible as it gets ported over.

Q-Are there going to be changes in the audio system?
A- Yeah it will still be Unity audio though. It has received improvements since the 1.X version of Unity. As far as the VaM UI interface and usability, everything is being overhauled to be less clunky.

Q-Would be nice to have two options, male or female, instead of just "person" , maybe three if you add "futa"
A- In 1.X you can set a default preset on the Person atom and then it will load as your preference every time when creating a new atom. For 2.X there will be separate atoms for male, female, futa, and other variants. Users will be able to make their own person atoms as well; they could be using custom grafts or even models outside of Daz3d. Unfortunately due to HD mesh and morphs, morphs will no longer be cross-gender compatible, and custom morphs in general will be harder to create due to how the import has to be done.

Q-will 2.0 be easily adapable to be NSFW? I'd like to use it for protyping and previz for for film and theatre.
A-Yeah I can release a SFW variant that uses models without gens. You probably don't want the Hub browser in there either.

Q-Will VaM scenes as a whole be exportable into unity for editing?
A-I'll have to think about the VaM scene to Unity question. There will be several Unity tools we will provide to community for 2.X creation process, mostly geared around making custom atoms and texture packs.

Q-I would like to know if VRFREE will be compatible with VAM2.0 if not maybe in the near future .... .
A- 2.X will support mainstream controls out-of-the-box. If something is popular enough we will add built-in support. If it isn't we will have an API that should allow controlling VaM from an externally built application.

That would be necessary since these devices tend to come with their own drivers and dlls that need to be bound into the application. So if the core doesn't have those drivers built-in you could not simply write a plugin. But 2.X will support external applications to allow for more niche use for those seeking it. If an industry-standard hand and finger tracking solution is created and this product supports it, then the 2.X native core could be updated to support it.

Q-Would VAM 2.x have more built-in shapes?
A-We could add more shapes, but 2.X will also support community-created custom atoms, so if something we provide is not there it could be easily made. One limitation will be how collision works. Physx only support primitive shapes and convex mesh colliders. We may further limit this in 2.X due to how skin physics needs to work. You may have to build the overall custom atom collision model using a set of primitive colliders (spheres, capsules, etc.).

Q- in the unity versions after 2019 they added shader graph and vfx graph but the vfx graph outputs 2 files per creation. Will there be a way for vam2 to recognize 2 files for 1 object somehow once vam2 version of importing is available?
A-I'm extensively using shader graph already in 2.X and that is how I plan to support custom shaders. I have not messed with VFX graph yet so I'm not sure how to answer that part.

Q-ultraleap released unity and unreal plugin
A-Excellent. I do plan to add support for this once VR hands are added.

Q-The people who make the Handy are interested in working with developers to support their hardware. They expressed some direct interest in VaM.
A-For most hardware, support will be added through our planned VaM API and publicly shared to the community so it can be improved upon. A lot of hardware requires custom dlls we do not want to bind into the main application as they change frequently and maintenance is a pain. So some may require a separate external software program that then talks to VaM through API. If the hardware is mainstream enough, we will add support natively. If the hardware does not require a custom dll that is bound into application, support may be added using plugin (but also using the VaM API) that can be loaded directly in VaM.

Q-Have you talked to qdot about Intiface/Buttplug?
A-No I'm a bit busy with other stuff. 🙂I'm not planning to do much with hardware myself. I'm only planning the API. I will probably contract somebody to work on some of this. I need to focus on core capabilities.The community might also be able to step up and provide it once API is ready (as long as it is capable enough).

VaM Studio

Q-You've got 12000 patrons - lets assume all in the middle tier just for the calcs, so 52k$ a month. Now if i got it right currently 2 developers are working on the project. I assume there are other larger expenses like upkeep for the hub, software licenses, etc - but iam just wondering, couldn't you fit one or two ppl more to the team?
A-There are more than 2 of us on the project. I'm not giving out specifics. I have also had rotating contractors for specific work, like for the Hub. Yes the project is bringing in a lot. I have actually tried to hire more developers in the past, but it is hard to convince the ones I want to join to join because they

already have very high paying jobs ($200K+) that they can talk about in public. 2.X is not currently in a place where having more developers make sense. I have structured 2.X in a way that it will make it easier to have more developers work on it. 1.X code was not set up well for that. Also - the job I left to work on this project was paying me over $200K per year, so it also has to be worth it for me. I'm probably work 2X as many hours as I did in my last job...that said I really enjoy working on this project, but it has taken a toll with the crazy hours. I have done long bursts of work before, and yeah I know I need to be careful to avoid burnout. I do plan to get normalized again once the 1st alpha is out. It is a big push.

Q- I know you guys are working super hard on 2.0 and know that your hard work is appreciated immensely by the community. I can imagine its a huge undertaking and software development is uniquely draining, you and the other devs are probably exhausted after working on this day after day. I am so excited to see the first incarnation, it's going to be my highlight of 2022!
A- Thank you! Yes 2.X is a huge undertaking! I'm working a bit too much right now, but I plan to take a day or two off after the alpha is out because I have been working 7 days a week, and most days well over 10 hours. That isn't sustainable.

Q- Meshed, would you ever consider turning this from a project to a fully fledged company? Maybe even a start up to potentially bring in investors?
A- I consider the project a "fully fledged company" already, but we are private and I am using contractors instead of direct employees for flexibility and to keep the operation simple to run. I prefer to stay as a private company to stay in control and keep it simple. I don't want to spend my days dealing with investors and employees. I want to spend my days doing what I am best at - developing software. If you mean do I plan on moving off Patreon to retail, the answer is yes at some point when 2.X is mature. My vision is for the VaM 2.X core to be a highly extensible system that can be added to by us as a company officially and also by the community for years to come. The core will be upgraded over time so there will hopefully never have to be a 3.X. It will just evolve as newer technologies are added. If we do the core right this should be possible.

Q- is there a Trello progress map for 2.x?
A- I haven't yet set up a Trello board for 2.X. We have some internal documents of plans that I may translate to a high-level Trello. I don't plan to go into specific details like what 1.X had until the project is much further along.

User Interface

Q- can you tell us about user interface design, i saw amazing UI creation tools in unity store, really hoping vam 2 will have modern design....some glassy blurry ui. Support for Every ui elements changeable via plugin?
A-The alpha is just using a throwaway UI and is pretty plain. The plan is to create UI that allows for skins and customization. Also will add language support. We haven't started on this. 2nd alpha will likely focus on these 3 things:
1. skin physics
2. parameter and save/load system
3. UI (both desktop and VR - they will be independent this time)

Rendering

Q-https://docs.unity3d.com/Packages/com.unity.render-pipelines.high-definition@12.0/manual/deep-learning-super-sampling-in-hdrp.html - any comment on the possibility of DLSS?
A-Yes 2.X will support DLSS. I have not yet tried to get it working, but I just moved the project to Unity 2021.2 recently since it was moved out of beta. I believe it is not that hard to add support for it.

Q- will you include a no compromise highest quality shaders for screenshooter and video recorder when fps doesn't matter?
A-Yeah - the alpha is actually shipping with the Ultra quality shader and materials. I created multiple shaders with some features cut back, and from what I have seen so far, it does not make much difference. The FPS is currently CPU limited on a 3080. That will certainly change as we add environments, GPU hair, cloth, multiple characters. We will also support custom shaders. They will need to be built on HDRP shader graph though, so limits apply since shader graph itself has some limits. There won't be custom lighting, etc. as that requires a different rendering pipeline and shader system. Also - raytracing could be turned on for shadows and reflections. That is a huge cost, but would be good for screenshots. I have also messed with URP, so there is possibility of 2.X running on lower end hardware in future.

Q-So hdrp compatibility is a must?
A-Yeah must be HDRP compatible for now.

Q-What kind of lighting 2.x will be supported . As vam some what revolve around daz so that means we may be seeing point , spot nd distance lighting ?
A-2.X is using Unity HDRP. It supports point, spot, directional, and a new kind of realtime area light. While 2.X currently uses Daz3d Genesis 8 models, there is not going to be anything stopping us or the community from adding other types of models. HDRI skies are in the alpha already for realtime global illumination. Custom HDRIs will be supported.

Q-not-rtx gpu will use HDRI GI and rtx-gpu will use raytraced GI right
A-I'm not promising any raytraced functions because they still have some issues and are performance hungry. You will get HDRI GI without any special GPU support. It is a part of HDRP core. I will attempt to support raytraced features when they become stable in HDRP. Shadows and reflections are the most likely to work.

Q- there is tool in daz by which we can turn any mesh into light emitting source which can be used as ghost lights
A- I can only support lights that Unity supports. Daz Studio is generally not set up for high performance real-time rendering so features that Daz Studio can do are not always going to be possible realtime. You will be able to make emissive materials for both clothing and skins in 2.X, but they won't cast actual light onto other objects. It will make things look like they are glowing though without the need for a light source.

Q-Skin shaders can be improved later, core mechanism must be meshed high priority right now.
A-I have actually spent most of this last week messing with shaders and I am trying to wrap that up. I want to at least get a decent baseline skin shader. The default HDRP Lit shader with subsurface scattering leaves a bit to be desired. I have added a more complex shader. The alpha will allow you to try a few of the different shaders to see for yourself (and compare performance). You are correct though that after this, I'll probably put shaders on the shelf for a while because there is a lot of other important core work that needs to happen next.

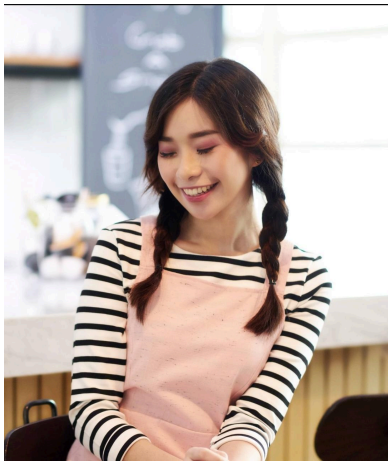
Q-not sure if asked before but how is 2.X planning to handle alphas? is it still going to use the render/shader queue method of 1.X or some new form of OrderIndependentTransparency like depth peeling?
A-HDRP has better support for transparent materials so it will be an improvement over 1.X fake transparency (rendered on opaque queue to allow shadows to work). I don't have exact details yet. You can read through some HDRP docs if you want to know more about what it supports.

Clothing & Hair

Q-Will 2.X have similar clothing system as Daz Studio? Like modifying clothes by regions and have smoothing modifiers etc.
A-2.X will have similar clothing to what you see in 1.X. We plan to add hybrid clothing that can have both rigid and flex/simulated portions. 1.X wrapped clothing smoothing could cause some issues with shrinkage of the cloth which would lead to clipping. Simulated cloth was not smoothed because sim cloth naturally smooths due to how it works with spring forces. It would be better to just import and run the high-def cloth than to bother with in-game subd

Q-Will clothing in 2.x be able to change softbody physics/morphs?
A-I don't know yet, but I hope to integrate clothing and soft-body physics so they can work together. I would like clothing to also be able to pull joints when appropriate. It won't be easy to accomplish. Hair too - pull hair should be able to pull the head down. Let's call those goals, not promises. 🙂

Q-Will braided hair & nicer ponytails be possible with the 2.x hair creator? Similar to this, where the hair is tied off and loose at the end of the braid.



A-Not likely to be able to braids like that using the hair sim. But 2.X will support custom rigged hair with a decent hair shader so what you show in that picture could be done that way instead.

Q-i'll just settle for cloth grab spheres keeping the cloth grabbed more easily on reload/save
A-The 1.X save system doesn't save the hair or clothing simulated state which is why cloth grab sphere doesn't really work properly on load. I will be seeing if it will be possible to save the clothing and hair vertex position and velocity state so that it loads exactly how it was saved. This might also lead to having clothing and hair state saves/presets if it works out.

Q-How does custom clothing factor into all this? Would clothing from a custom character be usable on other characters?
A-For clothing I'm going to try to see if the wrap part of clothing import can be done on-the-fly as needed to fit a custom character. That would make the clothing more portable. In some cases, stuff just won't be portable if the grafts prevent it from working well. Like if you tried to put a shirt on a model with wing grafts.

Q-does it mean it will always wrap to the closest morphed vert? The problem today is that some part of clothing wraps to the wrong vert (hoodies wrapping to the head which get destroyed when the head turns)
A-Yeah closest vert, probably on the unmorphed base, or maybe option to wrap to morphed. The problem as you mention with wrap is it doesn't really work well for vertices far away from skin. What is needed there is either rigid area that is not wrapped to vert but more traditionally skinned, or it should be simulated in that area. Simulation will be important to smooth out the clothing and prevent distortion. We will do a lot of experimenting to see if clothing can be improved. It really is not that easy of a problem to solve. 🙂

Q-Now when you say "2.X there will be separate atoms for male, female, futa," It makes me think that the content will be even more separated since now there will be 3 separate atoms?? If someone creates an outfit or hairstyle for the default female model for example, will it also be available for the default futa model automatically?
A- I hope to have a system for clothing and hair that allows them to work with all compatible atoms. I am also considering on-the-fly wrap which will try to fit it as best possible. If there are vast differences, things just won't work, like a shirt on a winged character (well it will work, but with clip through)

Q-my biggest peeve with at least 1.x is that you have no control over it once you wrap to morphed vertices (and sometimes not wrapping).
A-here are the following choices for clothing:
1. skinned (follows bones, does not comply to underlying mesh morphs or soft-body movement)
2. wrapped (follows mesh, adapts properly to morphs and soft-body
3. simulated (follows physics collision and tried to hold original clothing shape)
4. a mix of 1-3
1.X supports 2 and 3 and a mix of 2 and 3 by using a texture map mask. There really is no perfect solution until simulation gets so good it clothing just fits on the model like real clothing does.

Q-Regarding wrapped Clothing
A-Wrap works by taking each clothing mesh vertex and looking for the nearest skin triangle face and gets anchored to that. Then when the skin face moves (position/rotation) it moves the clothing vertex based on that. In 1.X you can have weight painting with the sim map to allow it to vary between the wrap position and the simulated position. This already exists. 2.X will likely offer a 3rd option: bone weighted (traditional skin), and additional combinations between the types. Not all clothing can work well with any specific type. Combinations will be needed. Some combinations are going to distort the clothing if the morphs on the model are too extreme for what the clothing was designed for. I can really only say this: 2.X clothing will be better than 1.X clothing in many regards, including simulation clip-through issues. I'm hoping you can run sim clothing in most cases and it will stick to the skin properly and handle movement properly.

Q-I'm asking about an approach to modify where each vertex wraps to the body (for example a long tails that only wraps to the base of the spine, or a helmet that only wraps to the head shape but doesn't wrap to the eyes lips nose etc. Since the wrap region copies both the morph and skinning at a particular vertex.

A-The best we can probably do is offer a mask to prevent wrapping to things like eyelids, etc. I did consider that in 1.X but never got to it.

Q-how about bone weighted cloth reacting to a morphed body part?
A-Yeah so you want bone weighted (for rigid) + auto adjust (or manual adjust) to account for morphs. I think we will be able to accommodate that in 2.X. I would like to support a wrap + bone combo. It will be tricky to determine how to offset the rigid areas to prevent clipping. I think in all cases it will require manual adjustment. I would like to see rigid armor with attached flexible (cloth) parts that stretches properly and avoids all clipping. The soft areas are likely going to work best as simulated instead of wrapped. That allows proper flex without clipping. Wrap can result in too many distortions and bad stretching. I don't have details on soft-body collision yet. I have some ideas but it is going to require experimentation to see what works best for performance and accuracy. The goal is mesh-accurate collision.

*Q-if morphed body is colliding with rigid armor then those *vertex will stop morphing. right?...otherwise they will clip through armor.*
A-Rigid armor is likely not going to be able to move based on collision. Unless you want to provide individual rigid pieces that can move as a whole.What I'm suggesting is you will have to fit the rigid clothing to a given morphed model using a set of adjustments. Once adjusted, it moves with the bones. Perhaps even physically tied to them to prevent joint movement when colliding. But the rigid part would not shift on the model itself.

Q-we can rig a tail which will have its own skeleton system or can we only use the body's skeleton?
A-You will be able to have rigged attachments with their own collision and joints.The hardest part of rigid clothing is not the rigid part, but the section that connects the rigid to the wrapped or simulated clothing. Rigid could be done with simple bone rigging and some morphs. I'm suggesting there could be a way to expand/retract rigid parts in specific ways to fit the morphed model without having to provide a different version, and have the wrapped/sim part auto stretch to fit. Wrap and sim already can handle morphs. Rigid (bone rigged) cannot. It could be that we come up with a way to handle rigid expansion (normal grow/shrink, etc.).

Q- will hair and cloth interact with all objects in 2.0?
A-That is the goal yes.

Animation

Q-Seems like animation was not discussed
A-Animation - will be a hybrid motion capture and keyframe animation system. It will have clips, patterns, sequences, etc. You will be able to edit motion capture and fix issues or merge pieces together. It will be one of the larger undertakings for 2.X. It will likely be built open-source style using the new API we will be creating.

Q-(editor's note: For brevity I'm linking to the start of a conversation thread regarding mimicking of natural skeletal and muscle dynamics within the framework of the VaM 2X engine. The full thread is a good read and worth your time. I'm going to focus on Meshed's responses here that are directly relevant to 2X and not quote the entire conversation. Be aware that the initial conversation continues into an animation specific sub thread that will be archived after 3+ days of inactivity.)

https://discord.com/channels/363274293112602636/909075760323629067/910896816936087573
A-There is a new system in 2.X for setting joint hold force and target angles that doesn't rely on the controls. The animation system could drive that, but it has error and that is why controls will still exist. People think of where they want to put hand and then it happens. Controls kind of mimic that by allowing to set a target position and rotation for endpoints. Joints in middle of rig are better handled by the joint target rotation. Animation systems often work on both foward and inverse kinematics, and 2.X will support both. Another thing I have experimented with is automated joint drive. You set a target and then a combination or joint target rotation + force to hold that position is applied. 2.X will be wide open for others to experiment with these systems and come up with unique drive systems. I would like to achieve "Active Ragdoll" trained by iterative learning that is becoming popular.

A-1.X is already sort of "IK". You set a point where you want the hand to go and it tries to match the position and rotation. The problem is it relies on the physics system to solve it and the solver being used has a lot of wiggle in it and competing forces.

A-In 2.X we can fix this somewhat with the new joint drive system. It could measure when the joint rotation target is far from the current position (i.e. the muscle is under stress) and it can adjust to minimize the force required.

A-VaM is meant primarily as a simulator. When you bump a leg, I want it to react in a realistic way. You can argue 1.X is not great in that regard and we want to improve that in 2.X. Sometimes I want to the character to "comply" with interactions (i.e. move the joint), but sometimes I want the character to hold the position. That means there has to be some way to communicate intention.

A- 2.X will improve animation system, but you can't expect to get a commercial animation engine. I would argue that physics assisted animation is going to look better for 99% of average users trying to make animations. I'll do the best I can at balancing these 2 things, and I'll make it so you could turn off or override the built-in systems when possible.

A-Most animation engines have some kind of IK otherwise it is a huge pain to do the animation. We are using the physics system as combination IK + some subtle physics with the goal of more realism while animating. It is not perfect for the reasons previously stated. You could turn joint drive to 0 and just let physics handle joints at the limits of their rotation, but the results will not be that realistic. The joint drive gives it more traditional IK feel. We are not currently planning on adding an additional IK system as that adds a whole other layer of complexity.

A-Portable animations are important for VaM, but VaM's main goal is to be a dynamic simulator, not an animation system for making 20 minute long fully animated scenes. It is best for shorter animations that can be dynamically modified to fit a situation (disable a node and take control using physics, etc.). We will make a better animation system in 2.X. This is all a bit premature as we won't be looking at this for some time. Nothing that is being asked for should be out of the question for 2.X. The animation system will likely be delivered as a plugin that can easily be modified and made better by the community.

Q- Acidbubbles is a cool guy, he even helped me back when I was working on my little scripting engine plugin
A- 2.X will have an animation system that allows both mocap and Timeline-like keyframe in one system. Yes it will be inspired by Timeline, and I'll be talking with AcidBubbles extensively about it. We have already discussed this. I even tried to hire @Acidbubbles multiple times, and will probably try again. 🙂 It

will get a real interface. But don't expect Blender, Maya, etc. That isn't happening. If you want to use a professional animation system, export the model rig (we will provide a way), animate externally, import the animation (we will provide a way), turn off physics drive, and now you have a perfectly good animation, but one you can't interact with. What I like about VaM is you can have subtle physics animations that are dynamic and changeable realtime.

Q-On the question of walking, it seems that having models walk around/follow you never seemed to have a good solution in 1.x. Was there something inherent about the 1.x code that made this difficult or would a similar challenge exist in 2.x?
A- The main challenge is VaM is set up as a physics simulator more than an animation engine. Walking implies a pre-canned walking animation with blended animations for turning, etc. Ideally we would have a system where the character can physically walk that accounts for stuff in the way and looks natural. It is not an easy problem to solve. I do think 2.X will get a locomotion system, but when the character physically hits other stuff it is going to break down.

Platforms and Support

Q- I wonder if making a port for Quest 2 (even a partial one) is under consideration for V2?
A- VaM is targeting realism, and for that you need compute power. I doubt I'll be able to get 2.X to run well on Quest 2, but I won't rule it out. It would require using a different rendering system than what 2.X is initially targeting (HDRP).

Q- Do you think VAMv1 could become open source once VAMv2 is established?
A- 1.X cannot become open source because I am using licensed assets that cannot be redistributed. I modified these assets heavily so people can't just purchase a license and get the correct functionality.

Q-Will V2 replace V1?
A-Until 2.X actually has a release where you can import custom content and has an animation engine, I don't really see it competing with 1.X. The alpha that is coming out soon will be more of a tech demo. You can't even save the scene.

Q-Will the game ever be on Steam?
A-When I feel that 2.X is to a state that is retail worthy (major features complete, easy-to-use, stable and bug free) I would consider Steam. I don't think doing a major marketing push right now is a good call. It would lead to a lot of new users who will probably be disappointed. 1.X has a steep learning curve and is no longer being updated. 2.X is in a state that is exciting, but isn't something you are going to spend a lot of time in until community features are added. Patreon is still a good fit for this stage of the project. Most people realize they are not paying for a product, but are helping fund the project that will get to a final product. 2.X is very early.