#	Question	Answer
1	What are the challenges in implementing AMD's gpu and rocm for hpc?	There is a bit of work with the codebase to make it work with the OpenCL API, but that is not too hard. We have found however, that compute speed seems to take a bit of hit with moving from the CUDA compiler to more generic OpenCL. So we haven't really worked seriously on the conversion yet. Another route would be to have a separate an optimised codebase for both CUDA and ROCm; the challenge there would of course be the need to maintain a far larger codebase, with a far smaller user group for the AMD side, at least initially.
2	What are the computer specs for groundwater modellers?	Hi Ashneel, the answer will vary a little depending on the size of the model you are running. Also, some model computer code is designed to parallelise better than others. I'm not personaly too familiar with MODFLOW however I can make the following general comments: If MODFLOW uses an explicit solution scheme it will likely benefit more from running on GPU or multiple CPU. If the solution scheme is implicit it may run faster on a modern CPU chip. On CPU you want a high CPU chip speed, sufficient CPU cache and fast SSD memory. You may need 64 to 128 GB of CPU RAM if the model is large. If the model can be parallelised than Jaap is running through key GPU components, i.e. recent cards wil have good GPU memory speed, and flops throughput. Note that 'flops throughput' is (very roughly speaking) a product of clock speed and number of cores - cores can perform operations in parallel and clock speed determines how often each core acts.
3	for example: MODFLOW-USG takes 10-15 hours to converg. what GPU power is needed?	I am not familiar with MODFLOW. Are you refering to power as in watts, or power as in compute capability? In general, there is no answer to what GPU would be needed, unless you set a specific target for how fast you need it to be. Then, benchmarking may help answer that question.
4	As a new hydraulic/flood modeller (less than two years), I've only used HEC-RAS (which seems to be using primarily CPU cores), I'm quite curious about other options. My company have been looking into moving more into cloud modelling (not just for hydraulic modelling) especially since many of the modellers are working remotely, what would you recommend? And is the transition from local computing to cloud relatively smooth or would each modelling software need to be ported to work with cloud computing?	(live answered) The transition from local to cloud can be very smooth, especially if you choose a form of computing in the cloud that is very similar to what you're used to locally. However, there are additional factors to consider like transport of models onto the cloud, and getting your results out. Also, the real strength of the cloud lies in parallelisation of many runs, and for that it's not so straightforward. You'll need to develop some knowledge and skills to get that to work, and it's not exactly the same across the various platforms. A general recommendation would be to find out what the optimum is for parallelisation (at what number of cores does your model run still gain sufficient peformance) and for CPU, there are clear limits to that. Of course 'GPU' and ' GPU' can differ greatly between engines, and the results won't be identical either, for more fundamental reasons - as discussed in the talk, consider your model and process before looking at hardware.
		HEC-RAS currently only use CPUs for computing currently. The program is currently be updated to for GPU compute. So in the future that this will be an option. I have heard a timeline of 1-2 year assuming things occur smoothly.
5	HI, when modelling a 1D/2D combined model, do the 1D calculation processed by CPU limit the use of the GPU for the 2D calcs? Are the speed gains from running 1D and 2D calculations on the GPU due primarily to memory/data access times? Is there a tipping point (1D vs 2D model size ratio) where the faster CPU clock speed outweighs the slower GPU clock speed even with the GPU's faster memory access?	To some extent. The 1D calculations cannot be done simultaneously with the 2D calcs, they have to be done sequentially. For models with thousands of 1D elements, the overall solution speed can drop a bit compared to running 2D only. We are currently working on getting a 1D engine on GPU so it is fast and it has access to the 2D data directly without having to copy it back and forth.
		That would be very useful and a significant performance increase when modelling urban catchments. Thanks
6	With the performance gains with HPC GPU, will TUFLOW Classic be phased out in the coming years, or limited future development?	We will continue to support Classic for a long time, but it is not being actively developed. HPC offers not just speed, but sub-grid-sampling and new turbulence models, both of which have been shown to be game-changes for solution convergence and accuracy.
7	What is a core?	A core is the smallest processing unit within the CPU that does calculations. Modern CPUs have up to 64 of them, greater numbers are coming in the near future or on servers. GPUs have thousands, but they work at a somewhat slower clock speed so it is hard to compare apples with apples here. Also, CPU cores are general purpose processing units, supporting a very broad and powerful instruction set, while GPU cores can only execute a very limited set of operations - but it just so happens that those are exactly the operations hydraulic models can benefit from. (in part, some of the instructions available on a GPU are more specific to graphics processing, or operate at higher precision than hydraulic models currently benefit from, which explains why not all GPUs are equally valuable)
8	Will there be consoderation for ARM processors? i.e NVIDIA grace hopper with high bandwidth memory	Good question. That is pretty cutting edge (and expensive) hardware. Once it becomes more mainsteam we'll take a look at it.

9	How would you define "improved Architecture" exactly? Is it simply more recent/updated CPU models?	Improved architecture is basically the hardware manufacturers coming up with better designs and thus new products that have better performance. Another way of looking at it would be calling them 'generations', so yes it is about more recent and updated, but we speak of a new architecture when the changes aren't just gradual or minor, but when structural improvement provide entirely new capabilities, or possibly break the way things were done in the past. For GPUs, new architectures come in rapid succession.
9	How would you define "improved Architecture" exactly? Is it simply more recent/updated CPU models?	Architecture is how the binary instructions are implemented by the cores and how the memory is managed and transfered. Newer architectures will typically run the same code faster even if using the same clock speed and same number of cores. Newer architecture may also offer support for entirely new functions.
9	How would you define "improved Architecture" exactly? Is it simply more recent/updated CPU models?	So we're specifically referring to the CPU architecture rather than the server farm architecture? We have laptops connecting to physical Workstations which connect to Servers. But your CPU focus implies that this architecture is less important than having a large number of recent CPU/GPU available. Note that 'architecture' is a broad term and can apply to any of these things - in this talk, architecture referred to the architecture of GPUs specifically, but the architecture of your compute systems, or network architecture can also have a profound impact on what the optimal process for running your models would be.
10	Are there any good rule of thumbs for estimating how much memory a model will require? I.e. x number of cells will require y number of memory?	Hi Samual, for TUFLOW moels there is some guidance here https://wiki.tuflow.com/Hardware_Selection_Advice - approximately 5 million cells can be run per gigabyte (GB) of GPU RAM and 4-6 times the amount of CPU RAM relative to GPU RAM (so 20-30GB of CPU RAM)
11	Due to the VDI sharing warning, is it better to run on bare metal rather than Virtualised?	Running on "bare metal" is simple and easy to implement, but requires multiple users to juggle rescources amongst themselves. If you only have a few users, that will work well, but for a larger team virtualisation is easier, but the issue is to do with how the GPU resources are shared. This aspect has to be done well. On Linux you may be able to use MIG and share resources to multiple VMs (which could be running either Linux or Windows). It's not necessarily better to run on bare metal - the VDI warning is specifically regarding time- and space-sharing on GPUs in ways where the model engine has no say over this. If you run on bare metal, you won't have issues, but you can set up good virtualisation options that make good shared use of GPUs, if you go the MIG route for example.
12	Is there a good way to estimate how much RAM will be required for initialisation of a given model setup under the latest build? i.e. is this a function of model cells and SGS resolution?	live answered, also see #10.
13	Would the bandwidth of a thunderbolt 4 connected external GPU be sufficient for tuflow HPC?	I really don't know. We have never tried it. HPC is quite careful with how much data has to be copied over the PCI bus, so it may in fact run OK on an external GPU. If ever you do get to test it, please send us your feedback!
		We have tested this when we moved a 3080 to an external enclosure (from a PCI4 x16 slot) to make way for a 4090. There was a ~20% drop in performance (increase in model time) which was surprising as typically going to a slower pci bus does not result in performance loss (with TUFLOW). The problem is that there can be many different factors impacting this - the raw bandwidth of the Thunderbolt 4 connection won't be the bottleneck, but other choices in the system architecture may affect this and you may see good results on one system, and worse on another. We can't recommend it in general, but it's not because it can't work in principle.
14	You say Storage: write locally. Would 8GB Fibre-channel connected SAN storage (directly connected, not mapped) be sufficient?	(live answered) The key thing is to write to storage that's 'local' to the machine. For a VM, that often means the storage isn't actually in the enclosure hosting the hypervisor running the VM, but somewhere like on SAN storage. But to the VM, the storage would be 'local'. The key takeway is that you should favour directly connected storage (within the limitations of the platform) to network-connected storage, like network shares or mounts.
		SAN via iSCSI? Im guessing you have VMs with GPU's attached to VMs? I'd say with sufficent IOPS/RW available + 10KSAS or SSD disks in your SAN, you should be fine. As Jaap mentioned, better CPU/GPU compute is ideal
15	Are you guys seeing any compatability issues with modern generations of Intel CPUs that use the big-little design (p-cores and e-cores) compared to say AMD Ryzen which is using a single core type?	We have noticed that when running on CPU it is possible to see very minor differences in the solution between Intel and AMD CPUs, but I should stress the differences are very small. In terms of speed, we haven't noticed that AMD performs vastly better than Intel or vice-verse - they are pretty similar. Note that your power use and heat production may differ between these two, but again this should be minor
		Cheers Greg. I've seen it cause issues with other engineering modelling. Plaxis3D for example did not like e-cores and we saw significant benifits disabling them.

16	By 1 cpu core, do you mean 1 thread or hyperthreaded cores with 2 threads? This looks to be modern standard for cpu nowadays	Hi Aaron, good question. When referring to cores we are typically speaking about the physical cores, each of which is typically virtualised to have 2 threads. I'll open this one up to Jaap later in the discussion also. When comparing hyperthreading vs. no hyperthreading the efficiency of running one vs the other has tended to vary a little depending on the model setup. Here's some tests from our coastal model https://fvwiki.tuflow.com/TUFLOW_FV_Parallel_Computing however this may not be as applicable for flood modelling. I'll open this one up for discussion at the end. (live answered) Because model runs are in part bottle-necked by memory access, they don't benefit as much from hyperthreading as a operations that can work mostly from cache or on the chip. However, it's complicated - in practice, we have found that for the TUFLOW engine hyperthreading provides very little to no benefit and you'll see the best performance if the number of threads in the model engine matches the number of available physical cores. So, we tend to mean 'physical core'.
17	With a RTX 4090 being roughly 35% faster in the benchmark model simulation to a RTX 4080, do you find that it's a better strategy to purchase the higher spec GPUs, as the cost associated with the time saved will offset the higher initial expenditure? Or do you find that you with the extra GPU power, you tend to create more detailed models, i,e, smaller cell size that end up with a similar run time?	Great question! Generally with the "gaming cards" it is worth the money to buy the best - especially when your licence time costs you money as well. Model size "creep" is real. But you do have to "keep up with the Jones'" - if others are running high-res models then you need to make sure you do not get left behind. Run your models at the resolution you want and need, but not higher. Getting faster and larger hardware shouldn't affect that once you're at the right size - but faster and larger hardware can save your engineers and consultants time and that quickly earns back the cost of the faster hardware. So, both, if you're not careful.
18	How much of the process involves writing to storage? Would this be a potential problem on HDDs for low grade HDDs that are not typically used for large amounts of writing/ re-writing	Hi Toby, this depends a bit on the model you are running. If you are writing very regular large outputs then file I/O tends to dominate and this will form the bottleneck. For typical models >500,000 cells I would expect much less than 5% of runtime is writing results as the majority of time is spent crunching numbers.
19	How much of a limitation is cooling - are you better off with maybe 1 less GPU and better cooling in a case vs stuffing as many GPUs as possible?	Hi Anon, thermal throttling can be a major issue. Personally we tried to put 4xRTX 3080s in a machine and we had to take one out because there wasnt' sufficient space and cooling capacity. This was a design issue in our box though as the cards were too close. If you get a good box, that is designed with sufficent cooling you should be ok and not suffer too much from throttling issues.
20	How are the Apple's M1, M2 (even M3 soon) GPUs? are they as near powerful as Apple says they are? or Is NVIDA the best way to go?	We haven't looked into them. We currently use NVidia's compiler tools, so TUFLOW HPC will only run on NVIdia hardware. If Apple find a way to support CUDA code then we may be able to test at some point in the future. Note that Apple M1 has 'Metal', which is different from both NVIDIA's CUDA and AMD's ROCm. Code written in OpenCL can be compiled to all three, but as explained previously, that involves a performance hit that model engines are currently not all willing to make.
21	What is HPC Speed exactly? How does it move information between components?	Hi Ben, it is typically measured in floating point calculations per second (FLOPS). The more calculations you can do per second the faster your model will run. These large HPC systems are also setup to ensure that information can move between the various components of the compute systems without being slowed down. For example the bandwidth of memory is fast enough that it doesn't become the 'critical path' or bottleneck slowing down the entire system.
22	As you know, we use Python in the JupyterLab platform to postprocess tuflowfv outputs. However, JupyterLab doesn't support network drives. Consequently, we need to perform post-processing tasks locally and subsequently move the scripts to a network drive accessible by other members of the project team. Is JupyterLab the only platform facing this limitation, or is there a way to run JupyterLab within network drives? Jupyterhub would be another option?	(answered offline) This is not really a hardware question, but it's important to keep in mind that any type of processing that performs a lot of reading and writing to storage (like post-processing typically does) is best executed on hardware that has a very fast connection to that storage. That often means that it should be 'near' the data. This problem is not specific to JupyterLab, and in fact you could access network storage from this platform just like you would from any Python-based platform.
23	What are quantum effects?	When transitions get really small, they will see unsual changes in their behaviour due to being built at a similar size scale as atoms. I'm not an expert on it, but it will be difficult to get die scales down to sub-nanometer. Generally, 'quantum' effects are effects that result from quantum-mechanical physical laws, whereas computer chips until now have mostly been based on classical physical laws governing electricity and heat. At large enough scale, quantum effects are negligible, but at these tiny scales, they start to dominate.
24	Since most bigger brands have gaming/personal and professional/industrial line-ups (Intel's Xeon and i series, Nvidia RTX and Quadro) which lines up better usually with modelling efforts, especially when comparing performance gains against additional cost?	Quadros are better as performance but aren't the 'bang for the buck' compared to the traditional card. The reason for this extra cost is often additional functions, or better precision that isn't needed for optimal performance of hydraulic engines. If you required the double precision (DP) version of the engine, you may benefit from a Quadro. If you typically use the single precision (SP) version - which is faster, but less precise, you're better off with GeForce (which includes RTX).
		Yep. There are way more gamers than engineers wanting to run models. I believe Nvidia also marks up their professional cards a lot more.
25	The best recommendation for CFD research?	Will depend on your CFD software. If it needs to run double precision then best to use a Quadro level card.
26	Have you seen any benifits to model runs with CPUs with larger L3 cache like the 5800X3D or 7800X3D CPUs?	We haven't noticed anything on this. Keep in mind through that when TUFLOW is running a medium to large model on GPU it is spending 95% of time ore more on GPU. So the CPU performance will make very little difference.

27	What would be a good way to spend money on AWS? start from small/less powerful specs then upgrade to a much higher spec when a bigger job arises?	Yes, focus on getting your workflow to work reliably, and figure out how you can scale up from running on one machine to running on several. AWS Batch is the right place too look, similarly Azure Batch for MS Azure is the same on that platform.
28	What is CUDA Code?	CUDA code is the code that runs on the GPU. It is different to the code that runs on the CPU. Note that when we say CUDA code we mean the code produced by the NVidia compiler to run on an NVidia GPU.
29	Cloud computing is still not built or tailored for hydraulic modelling, any good advice on future developments	live answered
30	Does tuflow scale across more than 2 GPU's effectively? we seem to notice a drop off on the third.	live answered
31	Does Tuflow have any future plans to have a SaaS or simplified PaaS based offering/Product for Tuflow platform modelling? Considering we are seeing many Software vendors create products on top of public cloud services. With $$ slapped on top of the public cloud costs.	Some other suppliers are stepping into this market, and TUFLOW is working to support them. The TUFLOW team itself currently primarily focuses on developing the engines, also in the direction that makes it easier to run them in these types of settings.
32	Thanks a lot guys! It looks like that local gaming PC is still the best for a individual engineer in a small company	That's not a bad conclusion - get the best you can afford, and make efficient use of it.
33	Awesome, thanks for your time, guys.	Thanks for coming Mark :)
34	Thank you for such webinars. Regards	You're welcome, please let us know what you think, and whether you'd want to see more like these.
35	This is a very important topic as we are engaged in Hydraulic Modelling at the moment.	Agreed. :)
36	Does GPU provide an advantage for running hydraulic models?	The specific computations required for hydraulic models lend themselves very well to parallelisation on GPUs, which means runtimes can be achieved on the GPU that far exceed anything possible on a CPU. Especially in a fixed grid model, or in other orthogonal configurations like QuadTree.
37	I want to know about the appropriate hardware to run these hydraulic models	You have come to the right place. :)
38	Should a single modelling computer be able to run more than one GPU model at a time?	Yes, if the GPU has sufficient memory for both models, and enough cores for there to be a benefit in running both models in parallel. There is no hard and fast rule about what these numbers are, but it is certainly worth trying. Also, if you used to be able to run a model on a GPU with x cores and y MB of memory, you should be able to run 2 of those models on a GPU with 2x cores and 2y MB of memory. Another limitation may be the surrounding hardware - as discussed in the webinar. For a single GPU, that will typically not be a concern.
39	When running HPC simulations, what are the main limiting factors for processing speeds? GPU, CPU, connection to servers, anything else?	As discussed in the webinar, your key consideration will be selecting the best GPU for your use case. However, a lack of CPU cores in a system with multiple GPUs, a lack of RAM to match the GPU RAM (4-6 times over) or poor/slow network connections to storage holding inputs and (more importantly) outputs are certainly factors to consider.
40	Good luck Greg and Jaap! Cheers, Simone	Thanks :)
41	HECRAS, GIS, & SMS	The Webinar was written from the viewpoint of TUFLOW developers and users, but we feel the conclusions translate well to any model engine that employs GPU compute, and we expect that in the near future, this will include most engines.
42	Cloud computing.	See #55
43	How do the relative software products compare on Intel vs AMD CPUs?	See #24
44	Please could you touch on: a) cloud data storage setup, b) transfer of large amounts of data back to local storage, c) do we keep raw data or just post-processed, and d) international security requirement differences, e.g. US vs EU?	These are all very relevant questions to answer when discussing model running on the cloud, but not so much particular to the subject of 'hardware'. Perhaps a similar webinar on 'running hydraulic models in the cloud' is warranted.
45	Looking for local hardware solution suggestions in Cloud-limited environment, preferably backed by benchmarking doco.	This was covered in the webinar; in short, have a look at the hardware benchmark results on tuflow.com, and consider comparing proxy benchmarks, like PassMark.
46	Want to know the specifications of computers	
47	Indicaran como se instala el software	No, the webinar is specific to hardware, however there are instructions on how to install the software and much more on the TUFLOW Wiki at wiki.tuflow.com