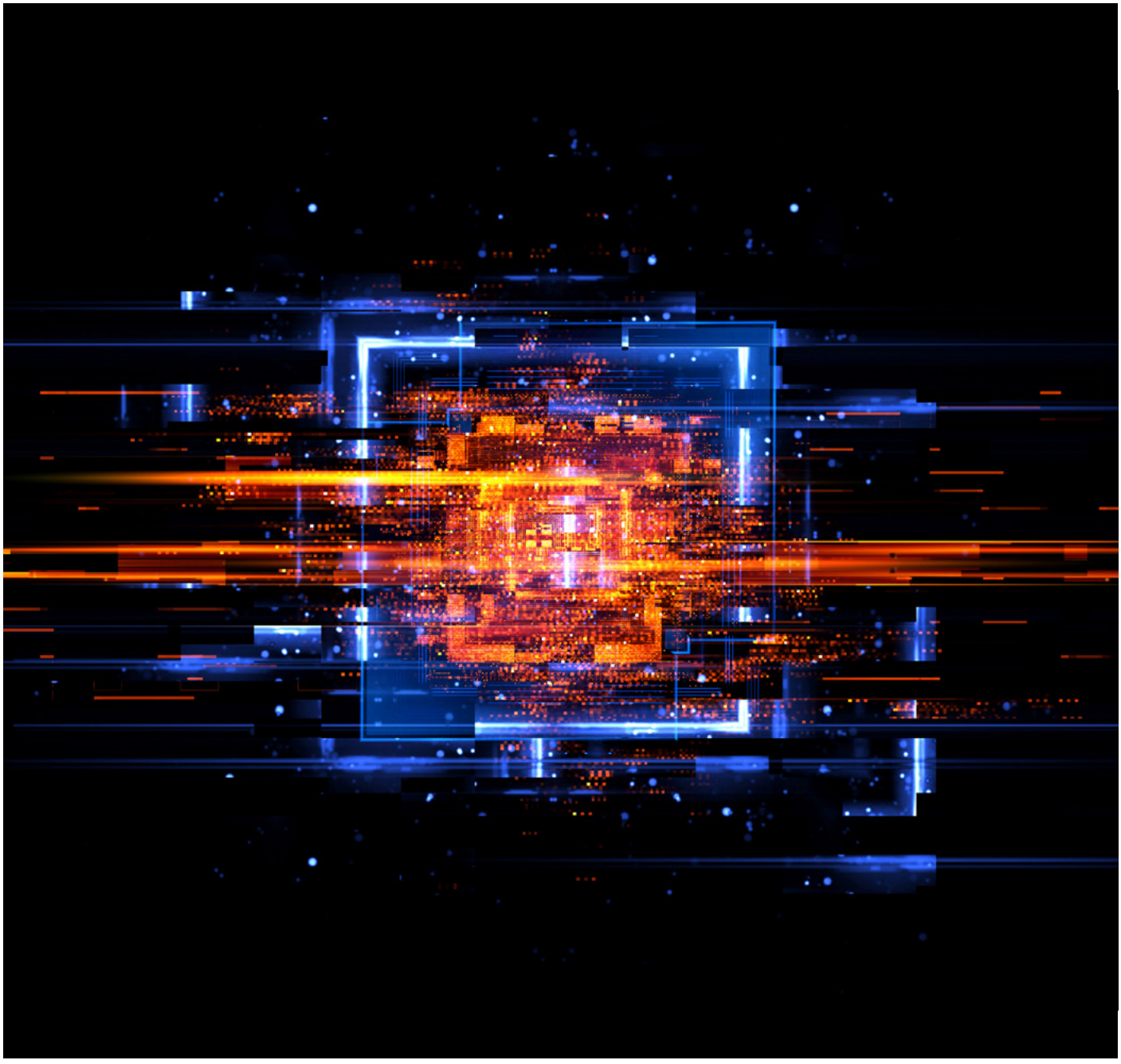
Taming the Exascale Beast: Challenges and Solutions in Architecting JUPITER

EVIDEN



Crispin Keable PhD. I am a senior solution architect developing HPC, AI and quantum systems for Eviden (an Atos business). My philosophy is and always has been to help customers achieve their scientific, technical, and engineering goals by using the best technology now and tomorrow in the most effective and efficient way. Today, that means the design and deployment of Exascale systems. While Exascale systems are in some senses one-off's, we can expect the technology and methods used to become widespread across the industry.

I work for Eviden, and in that capacity I have had the privilege to be the principle architect designing the JUPITER system for Jülich Supercomputer Centre (part of Forschungszentrum Jülich) as part of the EuroHPC JU initiative, providing the successful bid which will be implemented at JSC over the coming year. JUPITER will be the first exascale supercomputer in Europe, and this article is based on facing the complexities associated with designing and planning for the JUPITER system, and Eviden's approach to their solution.

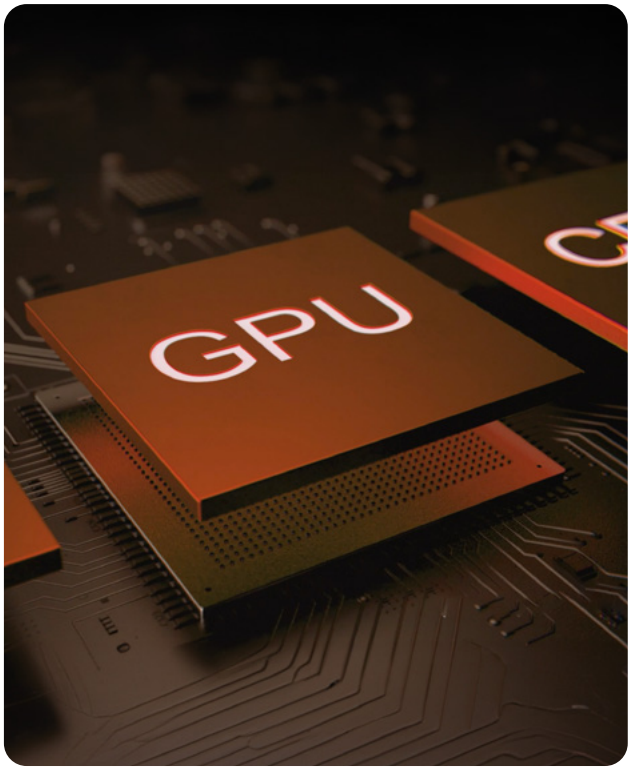
Supercomputing has always been something of an arms race, but since exascale (1 quintillion, 1018 calculations per second) has been on the horizon the race seems to have turned from a marathon into a sprint. HPC folk have been saying for a long time that to out-compete you have to out-compute, and recently governments around the world have been listening and taking action in the form of budgets to build these monster machines.

In this article, I don't intend to go into the drivers for exascale computing, on the assumption that they are clear enough to this audience – put succinctly, to be competitive a modern industrial economy needs a capable science and technology base, which in turn needs the tools to do its job. In this day and age, that means supercomputing – be it for simulation, machine learning, or some combination of the two.

Instead I want to discuss what is hard about computing at the exa level – we have had scalable systems for decades, why can't we just make them bigger and get to exascale?

While that is a single question, there are multiple answers, which all interlock. So I am going to try and give you my views on these topics, how they connect together, and what possible solutions there might be. The issues involved are power, density, complexity & RAS, software, and applications. All are important, there is a lot to be said about each of them, and the solution to any one really won't help: to make exascale genuinely useful we need to have a vision of solutions which stretch across all.

Exascale systems are large and very complex machines, built with tens of thousands of processors, connected by very high performance networking using thousands of cables which total to hundreds of kilometres in length, and can ship data around at rates that could move eleven thousand full copies of Wikipedia around in a single second. The networking is vital to the whole enterprise, since it allows complex science problems to be broken up across the system and computed, without having to wait for new data. If processors have to wait for their data, they just sit idle and there would be no real point in having built a machine that big in the first place.



The power issue arises because these systems are very big. If you built it just using current generation high end CPUs, it would take more than 100MW to power just the CPUs, ignoring all the DIMMs, switches, and storage you need. This is why scaling traditional HPC systems is a non-starter, and it is why the top end of the TOP500 list is dominated by accelerated systems, a trend expected to continue. For a sense of scale, 1MW solar power installation, good for a hospital or factory, would take the space of about 47 tennis courts. So you would need 4700 tennis courts worth of solar array to build an exascale machine like that – crazy! This is why accelerated machines are more and more the norm – with some arrangement of general-purpose CPUs, and then you offload the beefy calculations to GPUs. This brings the power cost down to more like 20MW for the whole system – still enough for a small town, but more tractable. Eviden has designed JUPITER in collaboration with NVIDIA, making use of the Grace-Hopper superchip.

Next, considering system density. Traditional data centre racks are air cooled, which can only be viable up to about 15kW per rack. This means that a 20MW system would need to be housed in over 1300 standard racks. By contrast, high end water cooling (such as we deliver in Eviden's BullSequana XH3000 system, used for JUPITER) increases density by about an order of magnitude, so the machine room only needs to house 130 racks.

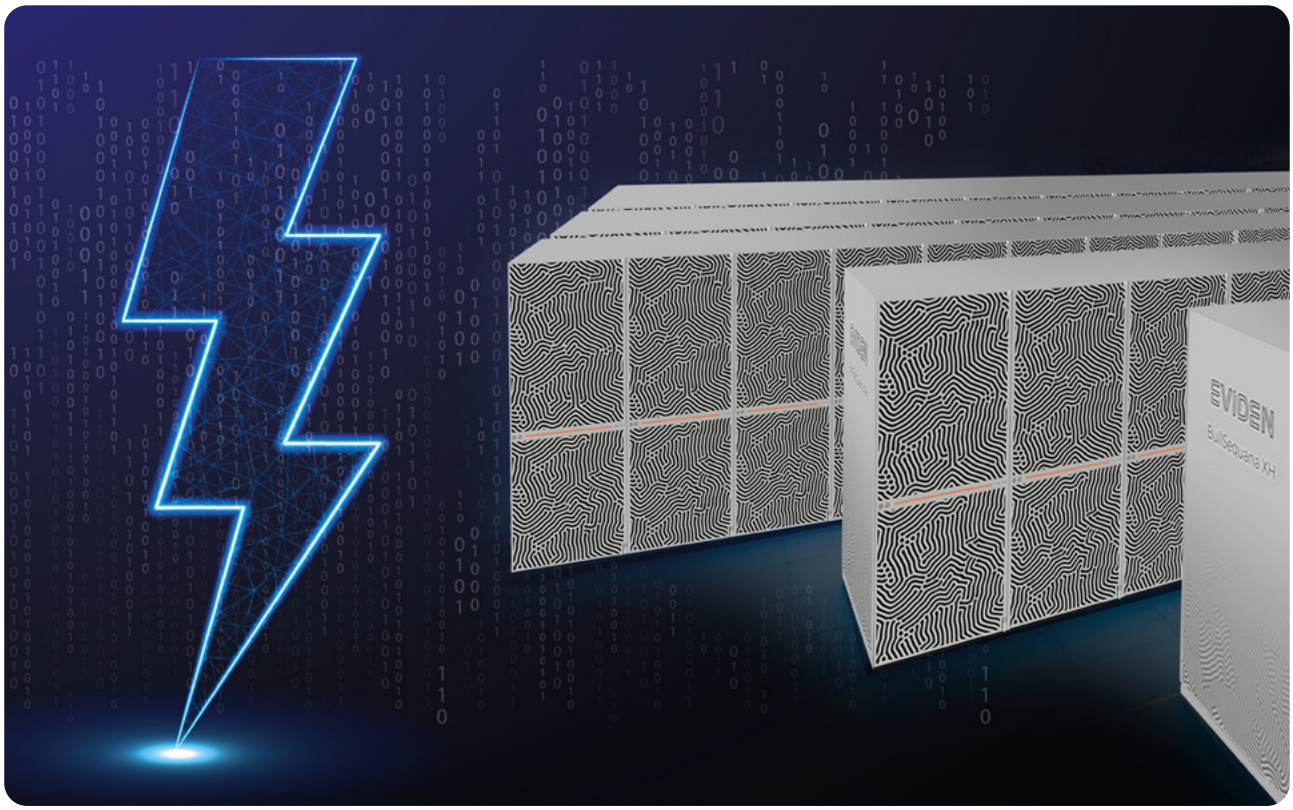
Cost of power is also important. With power costs as volatile as they are at the moment, keeping this type of machine running is both costly and unpredictable. To keep it running, you must pay for not just the electricity to make it function, but also the power used to cool the system. Traditional systems have given little thought to these costs, which means that in addition to many $10M's for power, you would need the same again for cooling. Once again this is where cooling becomes a vital factor, with hot water cooling contributing huge savings. With air cooling, it is not uncommon to pay as much again for cooling as you do for power, whereas with hot water cooling this can be reduced to a tiny fraction of the power cost, or even better the heat from the system can be re-used for district heating.

But what use is all this dumb machinery, without software that can tame, understand, and control it? An often-overlooked aspect of these machines is the software stack, on the erroneous assumption that you can use the same software already running on previous generations of machines, even desktops.

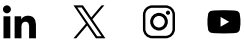
Researchers have made heroic efforts to parallelise applications for large scale systems in a wide range of application domains, and of course using these applications at exascale will deliver valuable new science. But at the same time, we will need new control software that can identify and work around failing components – machines are so big that there are likely to be some components failing at any time. We also need a better understanding of how to optimise applications for energy use, and control software that enables us to get the most science from machines for the least energy.

We will need a new generation of job control software – in the past, job control was like a giant game of Tetris trying to fit as much work into the machine as possible. But exascale computing has allowed scientists to work to wider ambitions, with applications working across much broader scales to solve problems like physically realistic models of the heart and brain, or nuclear fusion. Such models are typically large parallel applications, like before, but are parametrised as they run by a big gaggle of smaller models. These work at different length or time scales and often use completely different methods like machine learning. To get this whole edifice to run coherently, we will need a much more flexible approach to workload scheduling, one that takes into account the underlying science that is being computed.

In summary, the next generation of very large supercomputers such as JUPITER will be vital for transformational development across a wide range of science and technology disciplines, which in turn will have huge ramifications for society. Everything from health care, to new materials and methods for greening technologies, to clearer understanding and mitigation of climate change and much, much more will be transformed by these machines. But they aren't easy! Getting to that promised land of useful science from exascale is so much more than creating ever higher piles of kit, and then switching it on.



Connect with us



eviden.com