UberCloud Application Containers in the Advania Data Centers Cloud

This Compendium presents a collection of UberCloud case studies based on engineering simulations performed in the Advania Cloud, drawing from a select group of projects undertaken as part of the UberCloud Experiment and sponsored by Hewlett Packard Enterprise and Intel.

Advania Data Centers is a high-density computing technology company headquartered in Reykjavik, Iceland with operations in Sweden, Norway, Germany and the United Kingdom. Through extreme growth, Advania Data Centers now operate one of Europe’s largest datacenter campuses in Iceland that is tailor made for high density hosting such as High-Performance Computing (HPC), blockchain technology, high-density compute, all powered by renewable energy. Advania’s HPC team consists of experts that oversee the operation of HPC environments and HPC Jobs of their customers, globally leading organizations in manufacturing, technology, science among other industries. Advania partners with industry leaders in HPC such as Hewlett Packard Enterprise, Intel, Nvidia, and UberCloud to deliver next generation HPC environments such as HPCFLOW – Advania’s Bare Metal HPC Cloud, where HPC operators can execute in a fast and efficient manner.

UberCloud is the online community and marketplace where engineers and scientists discover, try, and buy Computing Power as a Service, on demand. Engineers and scientists can explore and discuss how to use this computing power to solve their demanding problems, and to identify the roadblocks and solutions, with a crowd-sourcing approach, jointly with our engineering and scientific community. Learn more about the UberCloud at: http://www.TheUberCloud.com.

In 2016, based on our experience gained from the previous cloud experiments, we reached an important milestone when we introduced our new UberCloud HPC software container technology based on Linux Docker containers. Use of these containers by the teams dramatically improved and shortened experiment times from an average of three months to just a few days.

Containerization drastically simplifies the access, use and control of HPC resources whether on premise or remotely in the cloud. Essentially users are working with a powerful remote desktop in the cloud that is as easy and familiar to use as their regular desktop workstation. Users don’t have to learn anything about HPC, nor system architecture, nor cloud to for their projects. This approach will inevitably lead to the increased use of HPC for daily design and development, even for novice HPC users, and that’s what we call democratization of HPC.

Wolfgang Gentzsch and Burak Yenier
The UberCloud, Los Altos, CA, June 2018

Please contact UberCloud at help@theubercloud.com before distributing this material in part or in full.
© Copyright 2018 TheUberCloud™. UberCloud is a trademark of TheUberCloud, Inc.
The UberCloud Experiment Sponsors

We are very grateful to our Compendium sponsors Hewlett Packard Enterprise and Intel, our Primary Media Sponsors Digital Engineering and HPCwire, and to our sponsors ANSYS, Autodesk, Microsoft Azure, Nephoscale, NICE DCV, and Comsol Multiphysics GmbH. Their sponsorship allows for building a sustainable and reliable UberCloud community platform:

Big Thanks also to our media sponsors HPCwire, Desktop Engineering, Bio-IT World, scientific computing world, insideHPC, and Primeur Magazine for the widest distribution of this UberCloud Compendium of case studies in the cloud:
## Table of Contents

3  **The UberCloud Experiment Sponsors**

5  **Team 169**: Complex Blood Flow through Cardiovascular Medical Device Using OpenFOAM on Advania

12  **Team 195**: Simulation of Impurities Transport in a Heat Exchanger Using OpenFOAM

16  **Team 196**: Development and Calibration of Cardiac Simulator to Study Drug Toxicity

20  **Team 197**: Studying Drug-induced Arrhythmias of a Human Heart with Abaqus 2017 in the Cloud

25  **Team 198**: Kaplan turbine flow simulation using OpenFOAM in the Advania Cloud

29  **Team 200**: HPC Cloud Simulation of Neuromodulation in Schizophrenia

35  **Team 202**: Racing Car Airflow Simulation with ANSYS Fluent on the Advania Data Centers Cloud

42  **Team 204**: Aerodynamic and CFD Simulations using Advania Data Centers’ HPCFLOW Technology

45  **Join the UberCloud Experiment**
MEET THE TEAM

End-User/CFD Expert: Praveen Bhat, Technology Consultant, INDIA
Software Provider: Lubos Pirkl, CFD Support, Prague, Czech Republic
Cloud Expert: Hilal Zitouni, Fetican Coskuner, Burak Yenier, The UberCloud, California, USA.

USE CASE
Many small and medium size manufacturers cannot afford to buy a powerful and expensive compute server to be able to run more complex and larger numbers of simulations which is necessary to manufacture higher quality products in shorter time. Buying a high-performance computer for the company means long procurement cycles, HPC expert knowledge to administer and operate the computer, additional expensive engineering software licenses, and high total cost of ownership.

This case study proofs that using Advania’s cloud resources together with UberCloud’s application software containers provide an excellent alternative to owning on-premise computing resources, coming with ease of software portability to the cloud, instant access to and seamless use of Advania cloud resources, and performance scalability from few to many cores, with an on-demand and pay-per-use business model.

During this one-week Proof of Concept (PoC) we used UberCloud containers to set up a technical computing environment on the Advania Platinum instances. OpenFOAM, the popular open source computational fluid dynamics toolkit was used to simulate complex blood flow through a cardiovascular medical device. Pre-processing, simulation runs, and post-processing steps were performed successfully with the OpenFOAM container coming with a fully equipped powerful virtual desktop in the cloud and containing all the necessary software, data and tools.

“The Advania Cloud and UberCloud containers are an ideal resource for our European engineering and scientific customers who want to burst into the cloud for complex resource-hungry HPC workloads.”
An Advania Qstack Cloud Service demo account was created and the Advania control panel was used as the primary method of infrastructure administration.

The Advania Cloud Services user interface was easy to use and responsive.

The Platinum 3X Large instance type was selected for the PoC due to its large size.

The Platinum 3x Large specifications are:
- 16 virtual CPU cores
- 61 GB RAM
- 20 GB disk was selected

The instance start time was around 2-4 minutes.

Instances were easily cloned and the clone instances performed as expected.

Figure 1: Advania Environment Setup.

An Advania Qstack Cloud Service demo account was created and the Advania control panel was used as the primary method of infrastructure administration. The Advania Cloud Services user interface was easy to use and responsive. The Platinum 3X Large instance type was selected for the PoC due to its large size. The Platinum 3x Large specifications are: 16 virtual CPU cores, 61 GB RAM, and 20 GB disk. The instance start times were around 2-4 minutes. Instances were easily cloned and the clone instances performed as expected.

Figure 2: Firewall configuration through Advania control panel.
DOCKER RUNTIME AND UBERCLOUD CONTAINER SETUP
The Advania instances were accessed via SSH, and the Docker run time environment was set up. This set up process took around 5 minutes and was automated down to a single command. The OpenFOAM container was pulled from the UberCloud private registry. This process took around 10 minutes. The OpenFOAM container was then launched on the Docker run time environment with no further configuration or set up. To allow access to the OpenFOAM container via remote desktop VNC service, the related ports were opened through the Advania control panel as seen in Figure 2.

THE ENGINEERING USE CASE: MEDICAL DEVICE MODEL

Figure 3: CAD model of the blood clot filter placed inside an artery. Captured using ParaView running inside an UberCloud Container on the Advania Cloud. To increase complexity of the problem blot clots were also inserted into the model (not shown above).

The model which was set up for testing is a simulation of blood flow through a cardiovascular medical device, a blot clot filter. Figure 3 shows a screenshot of the CAD model captured by accessing ParaView running inside an OpenFOAM container.

The CAD model of the medical device was used to generate a mesh of 5 million cells. The blood flow through the medical device was computed on 16 cores in parallel over 1,000 time steps using the simpleFOAM solver. The results were then post-processed using ParaView running inside the OpenFOAM software container.

Figure 4 shows the resulting streamlines, representing the path the blood flows in the artery and at the inlet of the medical device. Figure 5 displays the velocity plot, showing the blood flow at the inlet and the outlet of the medical device.
Figure 4: Streamlines, representing the path the blood flows in the artery and at the inlet of the medical device, using ParaView running inside an UberCloud Container on the Advania Cloud.

Figure 5: Velocity plot of the blood flow at the inlet and the outlet of the medical device, using ParaView running inside an OpenFOAM Container on the Advania Cloud.
USER EXPERIENCE

Figure 6: Full featured desktop view for remote visualization using ParaView in an UberCloud Container on the Advania Cloud. The desktop view provides usability and eliminates user training needs.

ParaView running inside an OpenFOAM Container on the Advania Cloud and accessed remotely via VNC demonstrated good performance. The end user was able to post-process the results, view 3D representations and manipulate these graphics (pan, zoom, rotate, etc.) in real time. The full featured desktop provided the entire regular feature set of ParaView (see Figure 6); there was no training required for the user to access and use this application container on the Advania Cloud.

Screen captures were generated using ParaView and the resulting image files were transferred from the UberCloud container to a local desktop using SCP to generate this report.

MONITORING AND PERFORMANCE

During the testing phase, system utilization information was collected through two methods: the Advania dashboard and the fully automated UberCloud Container Monitoring. Advania dashboard offers basic, easy to use monitoring of the CPU and network utilization. The report can be run for daily, weekly and monthly intervals. The reports update frequently and reflect the utilization of the resources at summary level.

UberCloud containers are equipped with an automated monitoring feature, which sends the user an up to date snapshot of the system utilization levels, and the progress of the running simulation. During testing the automated monitoring feature of the UberCloud containers running on Advania resources worked as expected and the testing team was able to monitor system utilization and record when test runs are
Technical Computing in the Advania Data Centers Cloud

This test was not intended to achieve the best performance possible; no effort was put into tuning the compute environment and gathering statistically relevant performance data. To provide a sense of the intensity of the calculation the following rough estimates are provided.

During the testing phase, system utilization information was collected through two methods: the Advania dashboard and the fully automated UberCloud Container Monitoring.

The Advania dashboard offers basic, easy to use monitoring of the CPU and network utilization. The report can be run for daily, weekly and monthly intervals.

The reports update frequently and reflect the utilization of the resources at summary level.

UberCloud containers are equipped with an automated monitoring feature, which sends the user an up to date snapshot of the system utilization levels, and the progress of the running simulation.

During testing the automated monitoring feature of the UberCloud containers running on Advania resources worked as expected and the testing team was able to monitor system utilization and record when test runs are complete.

This test was not intended to achieve the best performance possible; no effort was put into tuning the compute environment and gathering statistically relevant performance data. To provide a sense of the intensity of the calculation the following rough estimates are provided.

On a Platinum 3X Large instance, the SimpleFOAM solver ran on 16 cores in parallel for 1,000 time-steps of the cardiovascular device simulation in 30,000 seconds (roughly 8 hours).

**TOTAL EFFORT**

The total effort (without the 8 hours simulation run time) described above to access Advania’s OpenCloud, familiarize with the environment, setting up the OpenFOAM container, testing, developing the medical application geometry, boundary conditions, starting the jobs, and doing the post-processing with ParaView, and contacting and talking to Advania Support, was as follows:

- 2 hours in setting up the test account, getting familiar with GUI, requesting increase in quotas
- 1 hour in setting up the Docker environment, getting our base container, doing a quick test
- 3 hours in setting up the medical device simulation, doing steps like meshing, running the simulations (by the way, we ran it 5 times), monitoring, opening tickets with support, etc.

In total this resulted in a person effort of 6 hours for all the activities described above.
BUSINESS BENEFITS AND NEXT STEPS
The tests on Advania resources proved the compatibility of UberCloud’s technical container technology and Advania’s compute resources. Using the two together, a desktop like environment, with familiar user experience and with very low overhead can be set up, effortlessly. The major business benefits which are demonstrated by this case study are:

Benefits for the end-user:
- Portability: any cloud looks like the user’s workstation
- User-friendly: nothing new to learn, ease of access and use
- Control: container monitoring allows the user to control his assets in the cloud.

Benefits for the resource provider:
- Getting variability into their environment. Customers want different products which is easily implemented with container packaging and stacking
- Low overhead resulting in high performance
- High utilization by better use of resources.

Benefits for the ISV:
- Portability: software can run on a variety of different resource providers; built once, run anywhere
- Control of software usage via container based license and usage monitoring
- Control of user experience
- The faster the software runs the better the user experience; containers enable porting of the software to workstations, servers, and to any cloud.

Case Study Author – Praveen Bhat and Wolfgang Gentzsch
Team 195

Simulation of Impurities Transport in a Heat Exchanger Using OpenFOAM

Figure 1. 3D model picture of the initial version of heat exchanger, with coil inside.

“I’ve been using cloud computing for several years now, tried at least four different cloud providers and found the UberCloud service by far the best. I didn’t expect it would be SO easy to use.”

MEET THE TEAM

End-User/CFD Expert: Eugeny Varseev, Central Institute for Continuing Education & Training (ROSATOM-CICE&T.), Obninsk, Russia

Software Provider: Lubos Pirkl, CFD Support, with OpenFOAM in Box hosted in an UberCloud software container, Prague, Czech Republic

Resource Provider: Aegir Magnusson, Per-Ola Svensson, Hans Rickardt, Advania, Iceland


USE CASE

In this case study the numerical simulation of the impurities transport in a heat exchanger designed for coolant purification was performed using CFD Support’s OpenFOAM in Box v16.10 packaged in an UberCloud software container and hosted on the Advania Cloud. The transient process of the purification trap operation was simulated in to find the process stabilization time.

Almost any power equipment requires to maintain some level of coolant purity to provide the most reliable, effective way of operation. Studying the characteristics of the purification trap considered within this simulation is driven by the need to sustain the number of the impurities at a reasonably low level to keep the equipment of the circuit from fouling and heat transfer deterioration.

The study was performed at two general stages: first, the steady-state thermal hydraulic simulation of coolant flow pattern inside the heat exchanger was done using standard OpenFOAM capabilities on the local desktop. Second, the transient simulation of both dissolved impurities and crystalized particulates transport was performed using a custom OpenFOAM transport solver hosted in an UberCloud OpenFOAM software container.
METHOD

The simulation case was locally prepared on the engineer’s desktop and based on a CAD model created using the Salome software. Meshing was done by means of the snappyHexMesh utility. The model is a cylinder with an inlet tube inserted inside and an asymmetrically located outlet pipe at the top (see Figure 2). During the first stage of the study, which is computationally less demanding, a number of thermal hydraulic simulation runs were performed to determine the optimal mesh size of the model, which is between 0.1 M, 0.9M and 1.5M of hexahedral cells.

For the next stage, a custom OpenFOAM solver has been designed to consider the crystallization of dissolved impurity occurring due to coolant temperature decrease.

The formula of the impurity transport equation can be represented in the following mathematical way:

\[
\frac{dC_i}{d\tau} + \text{div}(uC_i) - \text{div} \left( \frac{\nu + \nu_t}{\text{Sc} + \text{Sc}_t} \text{grad}C_i \right) = Q_i,
\]

where \( C = \) impurity concentration, ppm; and index «i» means dissolved and crystallized phases; \( u = \) coolant velocity, m/s; \( \nu \) and \( \nu_t = \) viscosity and turbulent viscosity, m²/s; \( \text{Sc} \) and \( \text{Sc}_t = \) Schmidt number and turbulent Schmidt number; \( Q = \) source of concentration in the cell (dissolution of crystallization), ppm.

Figure 2. Symmetrical half of the CAD model, steady-state velocity field, and mesh of the model.
The custom computational model considers additional phenomena, such as:

- If the value of concentration in the given cell is less than that of the impurity concentration of the saturation \((C<C_s)\), the value of saturated impurity concentration is set equal to the saturation concentration and surplus concentration transforms to particulate phase with concentration \(C_p\).
- The reverse process of impurity dissolution.

The validation and verification of the custom solver based on experimental data of mass transfer in pipes preceded the simulation runs.

After the custom solver was ready to use, it was uploaded into the UberCloud container, precompiled for OpenFOAM v3.0, and moved into a folder for user solvers, and then it was ready to run right away.

**SIGNIFICANT CHALLENGES**
The stabilization time of the purification process is in the order of dozens of hours of real life time, so for transient simulation with time step value in the order of 0.001 sec using several millions of cells the purification simulation is definitely very time consuming. With the power of HPC, however, reducing simulation time dramatically, allows for studying models with less simplifications.

**RESULTS AND DISCUSSION**
The simulations were running on Advania cloud resources, on one dual-socket compute node with 2x Intel E5-2680 v3 processors, 24 cores, and 16GB of memory. The UberCloud software container on these resources allowed for getting at least a 15 times performance increase compared with the desktop system used for preparing this use case.

The spacial distribution of the dissolved impurity and particulates inside the purification trap was obtained as a result of the simulation. The analysis of the dissolved and precipitated concentration fields allowed obtaining mass transfer characteristics of the device.

The time it takes to stabilize the process was obtained from computation results by means of the ParaView post-processing and visualization right in the cloud and presented in Figure 3.

![Figure 3. Concentration at the outlet of the model via time.](image-url)
**BENEFITS**
The calculation in the cloud using an UberCloud OpenFOAM container allowed us to get the necessary result in rapid turn-around time – in less than 8 hours. This means a user can get his simulation results instead in days on his desktop now in just one night instead.

The ease-of-use experience was due to the fact that it takes virtually no time to adjust to the remote workspace offered by the UberCloud OpenFOAM container, because it looks and feels as if the user were doing the simulation on his personal Linux-based desktop. There are no special commands and configuration scripts to run.

The post-processing of files doesn’t require downloading big chunks of data back to the user’s computer – they simply were post-processed and analyzed right in the cloud by means of the tools the user is used to without any limitations. And for really big simulation cases this is especially important because they require huge computation power not only for the calculation but for the post-processing as well.

For file managing it’s possible to use conventional cloud resources, which are much more comfortable to use than FTP file managers, for example.

**CONCLUSION**
Transient simulation of purification trap device operation was performed in the cloud using CFD Support’s *OpenFOAM in Box v16.10* hosted in an UberCloud software container on Advania cloud resources. The time of the purification process stabilization was calculated with a 15 times computational performance advantage in comparison with the user’s personal desktop system used for the preparation of the use case.

The whole simulation process (mesh preparation, the simulation itself, and post-processing) has been done within the software container in the cloud, using automation and common post-processing scripts. It allows performing CFD studies and parametrical analysis of the models very quickly and as if a user just were using another workspace remotely.

**ACKNOWLEDGMENT**
Authors are very thankful to the IPPE Sodium laboratory team. Special thanks to F.A. Kozlov, Yu.I. Zagorulko, V.V. Alexeev, and C. Latge for helpful discussions of the topic.

---

*Case Study Author – Eugeny Varseev*
MEET THE TEAM

End User – Francisco Sahli Costabal, PhD Candidate, and Prof. Ellen Kuhl, Stanford University.

Software Provider – Dassault/SIMULIA (Tom Battisti, Matt Dunbar) providing simulation software Abaqus 2017.

Resource Provider – Advania Cloud in Iceland (represented by Aegir Magnusson and Jon Tor Kristinsson, with the HPC server from HPE).

HPC Cloud Experts – Fethican Coskuner and Wolfgang Gentzsch, UberCloud, with providing novel HPC container technology for ease of Abaqus cloud access and use.

Sponsor – Hewlett Packard Enterprise, represented by Stephen Wheat.

USE CASE

This experiment was collaboratively performed by Stanford University, SIMULIA, and UberCloud, and is related to the development of a Living Heart Model (LHM) that encompasses advanced electrophysiological modeling. The end goal is to create a biventricular finite element model to be used to study drug-induced arrhythmogenic risk.

A computational model that is able to assess the response of new compounds rapidly and inexpensively is of great interest for pharmaceutical companies. Such tool would increase the number of successful drugs that reach the market, while decreasing its cost and time to develop. However, the creation of this model requires to take a multiscale approach that is computationally expensive: the electrical activity of cells is modeled in high detail and resolved simultaneously in the entire heart. Due to the fast dynamics that occur in this problem, the spatial and temporal resolutions are highly demanding.

During this experiment, we set out to build and calibrate the healthy baseline case, that we will later use to perturb with drugs. After our HPC expert created the Abaqus 2017 container and deployed it on the cloud server, it was easy to debug and solve problems as a team. Also, sharing models and results between the end user and the software provider was easy.
HPE server in the Advania cloud, we started testing our first mesh. It consisted of roughly 5 million tetrahedral elements and 1 million nodes. Due to the intricate geometry of the heart, the mesh quality limited the time step, which in this case was 0.0012 ms for a total simulation time of at least 1000 ms. The first successful run took 35 hours using 72 CPU cores. During these first days, we encountered some problems related to MPI that were promptly solved by our HPC expert.

After realizing that it would be very difficult to calibrate our model with such a big runtime, we decided to work on our mesh, which was the current bottleneck to speed up our model. We created a mesh that was made out of cube elements (Figure 1). With this approach, we lost the smoothness of the outer surface, but we reduced the number of elements by a factor of 10 and increased the time step by a factor of 4, for the same element size (0.7 mm).

Additionally, the team from SIMULIA considerably improved the subroutines that we were using for the cellular model. After adapting all features of the model to this new mesh, we were able to reduce the runtime to 1.5 hours for 1000 ms of simulation using 84 CPU cores.

*Figure 1: tetrahedral mesh (left) and cube mesh (right)*

With this model, we were able to calibrate the healthy, baseline case, which was assessed by electrocardiogram (ECG) tracing (Figure 2) that recapitulates the essential features. Finally, we were also able to test one case of drug induced arrhythmia (Figure 3).
Technical Computing in the Advania Data Centers Cloud

**Figure 2: ECG tracing for healthy, baseline case**

**Figure 3: Snapshot of arrhythmic development after applying the drug Sotalol in 100x its baseline concentration.**

*The ECG demonstrates that the arrhythmia type is Torsades de Pointes.*
Some of the challenges that we faced were:

- Setting up the software to work with in Advania servers: there were a number of difficulties that appear due to the parallel infrastructure, the software that we used and the operating system. At some point, the system needed a kernel upgrade to stop crashing when the simulations were running. All these challenges were ultimately solved by the provider and HPC expert.
- The license server was at many points a limitation. In at least 4 occasions the license server was down, slowing down the process. Because all teams were in different time zones, fixing this issue could lead to delays in the simulations.
- Although the remote desktop setup enabled us to visualize the results of our model, it was not possible to do more advanced operations. The bandwidth between the end user and the servers was acceptable for file transfer, but not enough to have a fluid remote desktop.

Some of the benefits that we experienced:

- Gain access to enough resources to solve our model quickly in order to calibrate it. In our local machines, we have access to only 32 CPU cores, which increases the runtime significantly, making it hard to iterate over the model and improve it.
- As we had a dedicated server, it was easy to run post-processing scripts, without the need of submitting a second job in the queue, which would be the typical procedure of a shared HPC resource.
- Since all the people involved had access to the same containers on the servers, it was easy to debug and solve problems as a team. Also, sharing models and results between the end user and the software provider was easy.

Case Study Author – Francisco Sahli Costabal with Team 196.
Team 197

Studying Drug-induced Arrhythmias of a Human Heart with Abaqus 2017 in the Cloud

“We were able to easily access sufficient HPC resources to study drug-induced arrhythmias in a reasonable amount of time. With our local machines, with just 32 CPU cores, these simulations would have been impossible.”

MEET THE TEAM

End User – Francisco Sahli Costabal, PhD Candidate, and Prof. Ellen Kuhl, Living Matter Laboratory at Stanford University.

Software Provider – Dassault/SIMULIA (Tom Battisti, Matt Dunbar) providing Abaqus 2017 software and support.

Resource Provider – Advania Cloud in Iceland (represented by Aegir Magnusson and Jon Tor Kristinsson), with access and support for the HPC server from HPE.

HPC Cloud Experts – Fethican Coskuner and Wolfgang Gentzsch, UberCloud, with providing novel HPC container technology for ease of Abaqus cloud access and use.


“Our successful partnership with UberCloud has allowed us to perform virtual drug testing using realistic human heart models. For us, UberCloud’s high-performance cloud computing environment and the close collaboration with HPE, Dassault, and Advania, were critical to speed-up our simulations, which help us to identify the arrhythmic risk of existing and new drugs in the benefit of human health.”

Prof. Ellen Kuhl, Head of Living Matter Laboratory at Stanford University

USE CASE

This cloud experiment for the Living Heart Project (LHP) is a follow-on work of Team 196 first dealing with the implementation, testing, and Proof of Concept in the Cloud. It has been collaboratively performed by Stanford University, SIMULIA, Advania, UberCloud, and sponsored by Hewlett Packard Enterprise. It is based on the development of a Living Heart Model that encompasses advanced electro-physiological modelling. The goal is to create a biventricular finite element model to study drug-induced arrhythmias of a human heart.
Technical Computing in the Advania Data Centers Cloud

The Living Heart Project is uniting leading cardiovascular researchers, educators, medical device developers, regulatory agencies, and practicing cardiologists around the world on a shared mission to develop and validate highly accurate personalized digital human heart models. These models will establish a unified foundation for cardiovascular in silico medicine and serve as a common technology base for education and training, medical device design, testing, clinical diagnosis and regulatory science — creating an effective path for rapidly translating current and future cutting-edge innovations directly into improved patient care.

Cardiac arrhythmias can be an undesirable and potentially lethal side effect of drugs. During this condition, the electrical activity of the heart turns chaotic, decimating its pumping function, thus diminishing the circulation of blood through the body. Some kind of arrhythmias, if not treated with a defibrillator, will cause death within minutes.

Before a new drug reaches the market, pharmaceutical companies need to check for the risk of inducing arrhythmias. Currently, this process takes years and involves costly animal and human studies. With this new software tool, drug developers would be able to quickly assess the viability of a new compound. This means better and safer drugs reaching the market to improve patients’ lives.

The Stanford team in conjunction with SIMULIA have developed a multi-scale 3-dimensional model of the heart that can predict the risk of this lethal arrhythmias caused by drugs. The project team added several capabilities to the Living Heart Model such as highly detailed cellular models, the ability to differentiate cell types within the tissue and to compute electrocardiograms (ECGs). A key addition to the model is the so-called Purkinje network. It presents a tree-like structure and is responsible of distributing the electrical signal quickly through the ventricular wall. It plays a major role in the development of arrhythmias, as it is composed of pacemaker cells that can self-excite. The inclusion of the Purkinje network was fundamental to simulate arrhythmias. This model is now able to bridge the gap between the effect of drugs at the cellular level to the chaotic electrical propagation that a patient would experience at the organ level.

A computational model that is able to assess the response of new drug compounds rapidly and inexpensively is of great interest for pharmaceutical companies, doctors, and patients. Such a tool will increase the number of successful drugs that reach the market, while decreasing cost and time to develop them, and thus help hundreds of thousands of patients in the future. However, the creation of a suitable model requires taking a multiscale approach that is computationally expensive: the electrical activity of cells is modelled in high detail and resolved simultaneously in the entire heart. Due to the fast dynamics that occur in this problem, the spatial and temporal resolutions are highly demanding.
During the preparation and Proof of Concept phase (UberCloud Experiment 196) of this LHP project, we set out to build and calibrate the healthy baseline case, which we then used to perturb with different drugs. After creating the UberCloud software container for SIMULIA’s Abaqus 2017 and deploying it on HPE’s server in the Advania cloud, we started refining the computational mesh which consisted of roughly 5 million tetrahedral elements and 1 million nodes. Due to the intricate geometry of the heart, the mesh quality limited the time step, which in this case was 0.0012 ms for a total simulation time of 5000 ms. After realizing that it would be very difficult to calibrate our model with such a big runtime, we decided to work on our mesh, which was the current bottleneck to speed up our model. We created a mesh that was made out of cube elements (Figure 1). With this approach, we lost the smoothness of the outer surface, but reduced the number of elements by a factor of ten and increased the time step by a factor of four, for the same element size (0.7 mm).

![Figure 1: cube element mesh](image)

After adapting all features of the model to this new mesh with now 7.5 million nodes and 250,000,000 internal variables that are updated and stored within each step of the simulation (Figure 2), we were able to calibrate the healthy, baseline case, which was assessed by electro-cardiogram (ECG) tracing (Figure 3) that recapitulates the essential features.

![Figure 2: ECG tracing for the healthy, baseline case.](image)

During the final production phase, we have run 42 simulations to study whether a drug causes arrhythmias or not. With all these changes we were able to speed up one simulation by a factor of 27 which then (still) took 40 hours using 160 CPU cores on Advania’s HPC as a Service (HPCaaS) hardware configuration built upon HPE ProLiant servers XL230 Gen9 with 2x Intel Broadwell E5-2683 v4 with Intel OmniPath interconnect. We observed that the model scaled without a significant loss of performance up to 240 compute cores, making the 5-node sub-cluster of the Advania system an ideal candidate to run these compute jobs. In these simulations, we applied the drugs by blocking different ionic currents in our cellular model, exactly replicating what has been observed before in cellular experiments. For each case, we let the heart beat naturally and see if the arrhythmia is developing.
Figure 4: Evolution of the electrical activity for the baseline case (no drug) and after the application of the drug Quinidine. The electrical propagation turns chaotic after the drug is applied, showing the high risk of Quinidine to produce arrhythmias.

Figure 4 shows the application of the drug Quinidine, which is an anti-arrhythmic agent, but it has a high risk of producing Torsades de Pointes, which is a particular type of arrhythmia. It shows the electrical transmembrane potentials of a healthy versus a pathological heart that has been widely used in studies of normal and pathological heart rhythms and defibrillation. The propagation of the electrical potential turns chaotic (Figure 4, bottom) when compared to the baseline case (Figure 4, top), showing that our model is able to correctly and reliably predict the anti-arrhythmic risk of commonly used drugs. We envision that our model will help researchers, regulatory agencies, and pharmaceutical companies rationalize safe drug development and reduce the time-to-market of new drugs.

Some of the challenges that we faced during the project were:

- Although the remote desktop setup enabled us to visualize the results of our model, it was not possible to do more advanced operations. The bandwidth between the end user and the servers was acceptable for file transfer, but not enough to have a fluid remote desktop. We suggested to speed-up remote visualization which has now been implemented including NICE Software’s DCV into the UberCloud software container, making used of GPU accelerated data transfers.

- Running the final complex simulations first on the previous-generation HPC system at Advania took far too long and we would have not been able to finish the project in time. Therefore, we moved our Abaqus 2017 container seamlessly to the new HPC system (which was set up in July 2017) and got an immediate speedup of 2.5 between the two HPE systems.
Some of the benefits that we experienced:

- Gaining easy and intuitive access to sufficient HPC resources enabled us to study drug-induced arrhythmias of a human heart in a reasonable amount of time. With our local machines, with just 32 CPU cores, these simulations would have been impossible.
- As we had a dedicated 5-node HPC cluster in the cloud, it was easy to run post-processing scripts, without the need of submitting a second job in the queue, which would be the typical procedure of a shared HPC resource.
- Since all project partners had access to the same Abaqus 2017 container on the HPC server, it was easy to jointly debug and solve problems as a team. Also, sharing models and results between among the end user and the software provider was straight-forward.
- The partnership with UberCloud has allowed us to perform virtual drug testing using realistic human heart models. For us, UberCloud’s high-performance cloud computing environment and the close collaboration with HPE, Dassault, and Advania, were critical to speed-up our simulations, which help us to identify the arrhythmic risk of existing and new drugs in the benefit of human health.

Case Study Author – Francisco Sahli Costabal together with Team 197.

Appendix

This research has been presented at the Cardiac Physiome Society Conference in Toronto November 6 – 9, 2017, https://www.physiome.org/cardiac2017/index.html.

Title: Predicting drug-induced arrhythmias by multiscale modeling

Presented by: Francisco Sahli Costabal, Jiang Yao, Ellen Kuhl

Abstract: Drugs often have undesired side effects. In the heart, they can induce lethal arrhythmias such as Torsades de Points. The risk evaluation of a new compound is costly and can take a long time, which often hinders the development of new drugs. Here we establish an ultra high resolution, multiscale computational model to quickly and reliably assess the cardiac toxicity of new and existing drugs. The input of the model is the drug-specific current block from single cell electrophysiology; the output is the spatio-temporal activation profile and the associated electrocardiogram. We demonstrate the potential of our model for a low risk drug, Ranolazine, and a high risk drug, Quinidine: For Ranolazine, our model predicts a prolonged QT interval of 19.4% compared to baseline and a regular sinus rhythm at 60.15 beats per minute. For Quinidine, our model predicts a prolonged QT interval of 78.4% and a spontaneous development of Torsades de Points both in the activation profile and in the electrocardiogram. We also study the dose-response relation of a class III antiarrhythmic drug, Dofetilide: At low concentrations, our model predicts a prolonged QT interval and a regular sinus rhythm; at high concentrations, our model predicts the spontaneous development of arrhythmias. Our multiscale computational model reveals the mechanisms by which electrophysiological abnormalities propagate across the spatio-temporal scales, from specific channel blockage, via altered single cell action potentials and prolonged QT intervals, to the spontaneous emergence of ventricular tachycardia in the form of Torsades de Points. We envision that our model will help researchers, regulatory agencies, and pharmaceutical companies to rationalize safe drug development and reduce the time-to-market of new drugs.
Technical Computing in the Advania Data Centers Cloud

Team 198
Kaplan turbine flow simulation using OpenFOAM in the Advania Cloud

“Using the 24 cores available in the Advania Cloud allows up to 10 times faster calculations than our local computer and much more accurate simulation results”

MEET THE TEAM
End User – Martin Kantor, GROFFENG, a GRoup OF Freelance ENGineers.
Software Provider – Turbomachinery CFD based on OpenFOAM, Luboš Pirkl, Co-founder & Technical Director, CFD Support ltd.
Resource Provider – Advania Cloud in Iceland (represented by Aegir Magnusson and Jon Tor Kristinsson), with access and support for the HPC server from HPE.
HPC Cloud Experts – Fethican Coskuner and Wolfgang Gentzsch, UberCloud, with providing novel HPC container technology for ease of OpenFOAM cloud access and use.

About CFD Support
CFD Support supports manufacturers around the world with numerical simulations based on OpenFOAM. One of the main CFD Support’s businesses is providing full support for virtual prototyping of rotating machines: compressors, turbines, fans and many other turbomachinery. All the rotating machines need to be simulated to test, confirm or improve its efficiency, which has a major effect on its energy consumption. Each machine design is tested many times and is optimized to find the best efficiency point. In practice these CFD simulations are very demanding, because of complexity and number of simulations to run.

About GROFFENG
GROFFENG – GRoup OF Freelance ENGineers - is an open group of experienced Czech engineers focusing on Data analysis, Measurement, Data Acquisition and verification, Simulation and 3D Design. Not only do their engineers have experience and knowledge but they are also well equipped with hardware and software. This allow GROFFENG to provide quality and non-standard services to optimize technical processes.

USE CASE
This application can be found in the area of hydropower and the renewable energy sector. There are still many opportunities with usable hydro potential: existing hydropower plants with old obsolete turbines, new hydropower plants at an existing weir, or new hydropower plants for new locations.
Kaplan water turbines are used for locations with small head. For turbines with runner diameter 0.3-1 meters we can expect power 1 – 300 kW.

The flow simulation inside the turbine is calculated using the Turbomachinery CFD module (software by CFD Support) for OpenFOAM. The flow simulation and its analysis are important for the verification of turbine energy parameters, turbine shape optimization, and turbine geometry changes. Realistic application of the Kaplan turbine can be seen in the next picture.

Figure 1: Intake part of the turbine with guide vane and runner (left), two turbines during the installation (right).

Description of the turbine and simulation
Kaplan turbine for the low-head application includes the following: the inlet part with elbow and shaft, fixed guide wanes (blue color), runner with adjustable blades (red color) and conical draft tube.

Figure 2: Intake part of the turbine with guide vane and runner (left), two turbines during the installation (right).

The following turbomachinery settings are applied for these simulations:
- Turbomachinery CFD solver (software by CFD Support) includes MRF approach for rotation modeling;
- steady-state RANS simulations with k-omega SST turbulence models and incompressible water;
- time saving of the simulation is created by using periodic segment, each segment contains only one guide wane or runner blade;
- the boundary conditions are: volumetric flow rate for inlet, mixing plane for internal interface, cyclicAMI for periodic boundaries and fixed static pressure for outlet.
The computational mesh is created using snappy hex mesh algorithms (the mesh you can see in the following picture). For correct simulation flow inside the Kaplan turbine the following is important: uniform computational mesh of draft tube (for example in this case with inflation layers) and fine mesh in the gap between runner blade and runner camber (which you can see in the red cross section in the following picture). Our computational mesh with periodical segment has approximately 800k elements.

![Figure 3: Computational mesh for the Kaplan turbine with periodical segment with approximately 800k elements.](image)

The main task of this simulation is the calculation of energy parameters: i.e. the head and volumetric flow rate for defined runner speed of rotation and positioning of the runner blade. The task contained approximately 30 operating conditions, from minimal power output through best efficiency point to maximal power output.

Post processing is done with:
- global energy parameters using OpenFOAM scripts;
- flow visualization and analysis using ParaView (you can see velocity field and streamlines inside the turbine in the following picture).

**CLOUD APPLICATION AND BENEFITS**

The flow simulation is calculated using UberCloud’s Turbomachinery CFD container from CFD Support on up to 20 CPU cores on Advania’s HPC as a Service (HPCaaS) hardware configuration built upon HPE ProLiant servers XL230 Gen9 with 2x Intel Broadwell E5-2683 v4 with Intel OmniPath interconnect. Firstly, Martin Kantor had to prepare a geometry of turbine in local computer, the next step was the data transfer to cloud. The calculation settings were made from the previous calculation in local computer.

Turbomachinery CFD (TCFD) is a powerful tool for turbomachinery simulation with the computing mesh creation tool, the computation and evaluation process. You can see the TCFD GUI in the following picture: process bar is on the left, visualization of complete geometry is on the top in the middle, computational mesh with segments is on the bottom in the middle and convergence report is on the right.
Technical Computing in the Advania Data Centers Cloud

**Figure 4: The Turbomachinery CFD GUI.**

**Table 1: Time duration of simulation:**

<table>
<thead>
<tr>
<th>Platform</th>
<th>Time duration of 1000 iteration [minutes]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local computer (1 core)</td>
<td>90</td>
</tr>
<tr>
<td>Cloud application (2 cores)</td>
<td>80</td>
</tr>
<tr>
<td>Cloud application (4 cores)</td>
<td>34</td>
</tr>
<tr>
<td>Cloud application (20 cores)</td>
<td>20</td>
</tr>
</tbody>
</table>

The most effective strategy is to make several simultaneous simulations using 4 cores. Using the cloud (24 cores available) allows up to 10 times faster calculations than the local computer.

**BENEFITS OF USING CLOUD SIMULATIONS**

- high performance computing available at your fingertips;
- HW usage and all support are included in the cost for using the cloud service;
- simple and user-friendly operation of the cloud solution through the browser;
- possibility to perform postprocessing on the cloud or on the local computer.

*Case Study Author – Martin Kantor from GROFFENG, a GRoup OF Freelance ENGineers*
HPC Cloud Simulation of Neuromodulation in Schizophrenia

MEET THE TEAM

End Users – Dr. G. Venkatasubramanian, G. Bhalerao, R. Agrawal, S. Kalmady (from NIMHANS); G. Umashankar, J. Jofeetha, and Karl D’Souza (from Dassault Systemes).

Software Provider – Dassault/SIMULIA (Tom Battisti, Matt Dunbar) providing Abaqus 2017 software and support.

Resource Provider – Advania Cloud in Iceland (represented by Aegir Magnusson and Jon Tor Kristinsson), with access and support for the HPC server from HPE.

HPC Cloud Experts – Fethican Coskuner, Ender Guler, and Wolfgang Gentzsch from the UberCloud, providing novel HPC software container technology for ease of Abaqus cloud access and use.

Experiment Sponsor – Hewlett Packard Enterprise, represented by Bill Mannel and Jean-Luc Assor, and Intel.

USE CASE: NEUROMODULATION IN SCHIZOPHRENIA

Schizophrenia is a serious mental illness characterized by illogical thoughts, bizarre behavior/speech, and delusions or hallucinations. This UberCloud Experiment #200 is based on computer simulations of non-invasive transcranial electro-stimulation of the human brain in schizophrenia. The experiment has been collaboratively performed by the National Institute of Mental Health & Neuro Sciences in India (NIMHANS), Dassault SIMULIA, Advania, and UberCloud, and sponsored by Hewlett Packard Enterprise and Intel. The current work demonstrates the high value of computational modeling and simulation in improving the clinical application of non-invasive transcranial electro-stimulation of the human brain in schizophrenia.

Transcranial Direct Current Stimulation (tDCS): A new neurostimulation therapy

While well-known deep brain stimulation involves implanting electrodes within certain areas of the brain producing electrical impulses that regulate abnormal impulses, transcranial Direct Current Stimulation (tDCS) is a new form of non-invasive neurostimulation that may be used to safely treat a variety of clinical conditions including depression, obsessive-compulsive disorder, migraine, and...
central and neuropathic chronic pain. tDCS can also relieve the symptoms of narcotic withdrawal and reduce cravings for drugs, including nicotine and alcohol. There is some limited evidence that tDCS can be used to increase frontal lobe functioning and reduce impulsivity and distractibility in persons with attention deficit disorder. tDCS has also been shown to boost verbal and motor skills and improve learning and memory in healthy people. tDCS involves the injection of a weak (very low amperage) electrical current to the head through surface electrodes to generate an electric field that selectively modulates the activity of neurons in the cerebral cortex of the brain. While the precise mechanism of tDCS action is not yet known, extensive neurophysiological research has shown that direct current (DC) electricity modifies neuronal cross-membrane resting potentials and thereby influences neuronal excitability and firing rates.

Stimulation with a negative pole (cathode) placed over a selected cortical region decreases neuronal activity in the region under the electrode whereas stimulation with a positive pole (anode) increases neuronal activity in the immediate vicinity of the electrode. In this manner, tDCS may be used to increase cortical brain activity in specific brain areas that are under-stimulated or alternatively to decrease activity in areas that are overexcited. Research has shown that the effects of tDCS can last for an appreciable amount of time after exposure.

While tDCS shares some similarities with both electroconvulsive therapy (ECT) and transcranial magnetic stimulation (TMS), there are significant differences between tDCS and the other two approaches. ECT, or electroshock therapy, is performed under anaesthesia and applies electrical currents a thousand times greater than tDCS to initiate a seizure; as such, it drastically affects the functioning of the entire brain and can result in significant adverse effects, including memory loss. By contrast, tDCS is administered with the subject fully conscious and uses very small electric currents that are unable to induce a seizure, constrained to the cortical regions, and can be focused with relatively high precision. In TMS, the brain is penetrated by a powerful pulsed magnetic field that causes all the neurons in the targeted area of the brain to fire in concert. After TMS stimulation, depending on the frequency of the magnetic pulses, the targeted region of the brain is either turned off or on. TMS devices are quite expensive and bulky which makes them difficult to use outside a hospital or large clinic. TMS can also set off seizures, so must be medically monitored. By contrast, tDCS only affects neurons that are already active—it does not cause resting neurons to fire. Moreover, tDCS is inexpensive, lightweight, and can be conducted anywhere.

HPC BRAIN SIMULATION IN THE ADVANIA CLOUD
The National Institute of Mental Health and Neuro Sciences (NIMHANS) is India’s premier neuroscience organization involved in clinical research and patient care in the area of neurological and psychiatric disorders. Since 2016, Dassault Systemes has been collaborating with NIMHANS on a project to demonstrate that computational modeling and simulation can improve the efficacy of Transcranial Direct Current Stimulation (tDCS), a noninvasive clinical treatment for schizophrenia. Successful completion of the first stage of this project has already raised awareness and interest in simulation-based personalized neuromodulation in the clinical community in India.

Although effective and inexpensive, conventional tDCS therapies can stimulate only shallow regions of the brain such as prefrontal cortex and temporal cortex regions. These therapies cannot really penetrate deep inside the brain. There are many other neurological disorders which need clinical interventions deep inside the brain such as thalamus, hippocampus and subthalamus regions in Parkinson’s, autism, and memory Loss disorders. The general protocol in such neurological disorders is to treat patients with drugs and in some cases, patients may be recommended to undergo highly invasive surgeries. This would involve drilling small holes in the skull, through which the electrodes are inserted to the dysfunctional regions of the brain to stimulate the region locally as shown in Figure 2. This procedure is called as “Deep Brain Stimulation”, in short DBS. However, DBS procedure
has potential complications such as stroke, cerebrospinal fluid (CSF) fluid leakage, bleeding, etc. Other drawbacks are that not every patient can afford DBS surgery considering their individual health conditions and high cost medical procedures.

Figure 2: invasive surgeries involve drilling small holes in the skull, through which the electrodes are inserted to the dysfunctional regions of the brain to stimulate the region locally.

Our project demonstrates an innovative method that can stimulate deep inside the brain non-invasively/ non-surgically, using multiple electric fields applied from scalp. This procedure can precisely activate selective regions of the brain leaving minimal risk and also making it affordable to all.

Background
The method that is adopted here is called “Temporal Interference” (TI), where we are forcing two alternating currents (transcranial Alternating Current Stimulation: tACS) at two different high-frequency electric fields towards the brain via pairs of electrodes placed on the scalp. Neither of the individual alternating fields is enough to stimulate the brain because the induced electric field frequency is much higher than the neuron-firing frequency; hence the current simply passes through tissue medium with no effect. However, when two alternating current fields intersect deep inside the brain, a pattern of interference is created which oscillates within an ‘envelope’ at a much lower frequency i.e. difference between two high-frequencies, which is commonly referred to as “beat frequency”, which would stimulate a neural activity in the brain. With this method clinicians can precisely target regions of the brain without affecting major part of the healthy brain!

It is anticipated that “Temporal-Interference” stimulation has great potential to treat a large number of neurological disorders. However, it is required to be personalized for an individual depending upon type of disease targeted and inter-individual variation in brain morphology and skull architecture. Since each patient’s brains can be vastly different, an optimal electrode placement needs to be identified on the scalp in order to create Temporal-Interference at specific regions of the brain for an effective outcome. For instance, in Parkinson’s disease, thalamus and globus pallidus would most likely be the regions to create Temporal-Interference to regulate electrical signals and there by activating neurons to reduce the tremor in the patients.

The power of multi-physics technology on the Advania Cloud Platform allowed us to simulate the Deep Brain Stimulation by placing two sets of electrodes on the scalp to generate Temporal-Interference deep inside the grey matter of the brain, as presented in the Figure 3 workflow. However, a basic level of customization in post processing was required in making this methodology available to the clinician in real time and also reduce overall computational effort, where doctors can
choose two pre-computed electrical fields of an electrode pair to generate temporal interference at specific regions of the grey matter of the brain. Nevertheless, the technique proposed here can be extended to any number of electrode pairs in future.

![Flowchart](image)

**Figure 3: The workflow for the Virtual Deep Brain Stimulation on a human head model.**

A high fidelity finite element human head model was considered including skin, skull, CSF, sinus grey & white matter, which demanded high computing resources to try various electrode configurations. Access to HPE’s Cloud system at Advania and SIMULIA’s Abaqus 2017 code in an UberCloud software container empowered us to run numerous configurations of electrode placements and sizes to explore new possibilities. This also allowed us to study the sensitivity of electrode placements and sizes in the newly proposed method of Temporal-Interference in Deep Brain stimulation which was not possible before on our inhouse workstations and HPC systems.

The results demonstrated in the Figure 4 is for two sets of electrical fields superimposed to produce “Temporal Interference”:

- Configuration-1: Electrical fields generated from electrodes placed on the left and right side of pre-temporal region of the scalp.
- Configuration-2: Electrical fields generated from electrodes placed on the left of the pre-temporal and rear occipital region of the scalp.

In Configuration-1, the “temporal interference” was observed at the right hippocampus region, whereas for Configuration-2, the temporal interference” was observed at the subparietal sulcus.
Figure 4: The results show the sensitivity of the temporal-interference region deep inside the brain based on electrode placement on the scalp.

Based on this insight, the team is now continuing to work towards studying various electrode placements in targeting different regions of the brain. While preliminary results look promising, the team will be working closely with NIMHANS in validating the method through further research on this topic and experimentation. In parallel, the team is also working towards streamlining the methodology such that it can easily be used by clinicians.

HPC Cloud Hardware and Results
We ran 26 different Abaqus jobs on the Advania/UberCloud HPC cluster – each representing a different montage (electrode configuration). Each job contained 1.8M finite elements. For comparison purposes, on our own cluster with 16 cores, a single run took about 75min (solver only) whereas on the UberCloud cluster a single run took about 28min (solver only) on 24 cores. Thus, we got a significant speedup of about 2x running on UberCloud.

Figure 5: Localization of the peak Electrical Potential Gradient value in Abaqus for different combinations of electrodes.
CONCLUSION
In the recent times, the Life Sciences community has come together better than ever before, to collaborate and leverage new technologies for the betterment of health care and improved medical procedures. The application discussed here demonstrates a novel method for "Deep Brain Stimulation" in a non-invasive way which has the potential to replace some of the painful/high risk brain surgeries such as in Parkinson’s disorders.

The huge benefits of these computational simulations are that they (i) predict the current distribution with high resolution; (ii) allow for patient-specific treatment and outcome evaluation; (iii) facilitate parameter sensitivity analyses and montage variations; and (iv) can be used by clinicians in an interactive real-time manner.

However, there is still a lot of work to be done in collaboration with the Doctors/Clinicians at NIMHANS and other Neurological Research Centers on how this method can be appraised and fine-tuned for real time clinical use.

Case Study Authors – G. Umashankar, Karl D’Souza, and Wolfgang Gentzsch
Team 202

Racing Car Airflow Simulation with ANSYS Fluent on the Advania Data Centers Cloud

“UberCloud containers turns Advania’s HPE Infrastructure as a Service platform into a highly productive Software as a Service platform which was a great pleasure to work with!”

MEET THE PROJECT TEAM

End User – Praveen Bhat, Technology Consultant, India
Software Provider – ANSYS
Resource Provider – Jon Thor Kristinsson, Ómar Hermannsson, and Aegir Magnusson from Advania Data Centers
Technology Experts – Fabrice Adam and Andrew Richardson, HPE, and Reha Senturk, Ender Guler, and Ronald Zilkovski from UberCloud.

USE CASE

This aerodynamic study provides the air flow and the forces acting on a racing car to understand the air velocity and its impact on the car’s stability during racing. The study focusses on understanding the aerodynamic performance and quantifying different forces acting on the racing car at certain speeds. The Computational Fluid Dynamics (CFD) analysis provides an in-depth insight on the air flow, pressure and velocity distribution around the car and also parameters required to calculate the aerodynamic forces. 3D CAD model of the racing car with a dummy driver is modelled as part of this project. The CFD models were generated within the ANSYS 19.0 simulation environment on a max 128-core HPC compute cluster with 250 GB RAM, accessed using a VNC viewer through the web browser. ANSYS Fluent was running in UberCloud’s HPC application software containers on Advania Data Centers HPCFLOW cloud resources. The following flow chart defines the ANSYS container setup and modelling approach for setting up and running the simulation in the Ansys Containerized environment:

![Diagram of container environment with Ansys Fluent application](Figure 2)
The model construction and setup are done accordingly as shown in the following flow chart:

The following describes the step by step approach for setting up the CFD model within the ANSYS Workbench 19.0 Environment:
1. Generate the 3D racing car model with dummy driver with ANSYS Design Modeler. Air volume is modelled around the racing car for external flow simulation.
2. Develop the CFD mesh model for the 3D racing car with the surrounding air volume. Create the groups from the mesh faces for applying the boundary conditions. Save the file as a Fluent case file (*.cas file).
3. Import the CFD model into the Ansys Fluent environment. Define the number of cores needed to build and run the CFD simulations.
4. Define the Model parameters, fluid properties, and boundary conditions.
5. Define solver setup & solution algorithm, mainly related to define the type of solver, convergence criteria and equations to be considered for solving the external flow simulation.
6. Extract the pressure load on the racing car which is used for calculating the forces on the racing car and evaluate its stability under aerodynamic forces.

The ANSYS Fluent simulation setup is solved in the HPC Cloud environment. The simulation model needs to be precisely defined with good amount of fine mesh elements around the 3D racing car geometry. The following snapshot highlights the racing car geometry considered and the 3D Fluent mesh model:
Figure 5 shows the pressure distribution result at the mid-section of the 3D racing car. The pressure distribution across the section is uniform. The velocity plot in Figure 6 shows that the air...
velocity varies near the leading edge of the racing car. The air particle velocity is uniform with particles following a streamlined path near the car wall.

**HPC Performance Benchmarking**

The External flow simulation is carried out in Advania’s HPCFLOW cloud environment running on a 128-core server with CentOS Operating System and ANSYS Workbench 19.0 simulation package. The server performance is evaluated by submitting the simulation runs for different numbers of elements. Obviously, the finer the mesh size the more time is required to run the simulation. The run time can be minimized by using more cores. The following tables highlight the solution time captured for the 128-core system with element numbers of up to 140 million elements.

**Table 1: Simulation performance time (sec) for different number of cores**

<table>
<thead>
<tr>
<th>Model Size</th>
<th>No of Nodes</th>
<th>Cores for each node</th>
<th>No of cores</th>
<th>Solution time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>14 million elements</td>
<td>1</td>
<td>32</td>
<td>32</td>
<td>1049.35</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>32</td>
<td>64</td>
<td>531.89</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>32</td>
<td>96</td>
<td>361.11</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>32</td>
<td>128</td>
<td>279.37</td>
</tr>
</tbody>
</table>

**Table 2: Simulation performance time (sec) for different number of cores**

<table>
<thead>
<tr>
<th>Model Size</th>
<th>No of Nodes</th>
<th>Cores for each node</th>
<th>No of cores</th>
<th>Solution time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>140 million elements</td>
<td>1</td>
<td>32</td>
<td>32</td>
<td>16300.00</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>32</td>
<td>64</td>
<td>7530.60</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>32</td>
<td>96</td>
<td>5254.60</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>32</td>
<td>128</td>
<td>4251.29</td>
</tr>
</tbody>
</table>

**Figure 8: Runtime (secs) vs number of cores for the 14-million mesh model**
The simulation time reduces considerably with the increase in the number of CPU units. The solution time required for 32 cores with fine mesh model is 3.9 times higher than the time required for a 128-core server with the same mesh model. For a moderate number of elements (~14 million), the 32-core server performance is 4.5 times better than a normal Quad core system with respect to total number of simulation jobs completed in a day.

**Person-hours efforts Invested**

**End user/Team Expert:** 120 hours for setup, technical support, reporting & overall management of the project.

**UberCloud support:** 15 hours monitoring & administration of host servers and UberCloud containers, managing container images (building & installing container images during any modifications/bug fixes) and improvements, such as tuning memory parameters, configuring Linux libraries, and usability enhancements. Most of this is one-time effort and will benefit future users.

**Resources:** 1000 core-hours used for performing various iterations of this cloud simulation project.

**CHALLENGES**

The project started with setting up ANSYS 19.0 workbench environment with Ansys Fluent modeling software on one 32-core server. Initial working of the application was evaluated and the challenges faced during the execution were highlighted. Once the server performance was enhanced based on the feedback, the next level of challenge faced was scaling the existing system to a multi-node container environment where the ANSYS container used the scaled computation environment. The key challenge in the project was technical which involved accurate prediction of air flow behaviour around the racing car for the aerodynamic forces which was achieved by defining appropriate element size for the mesh model. The finer the mesh the higher the simulation time required and hence the challenge was to perform the simulation within the stipulated timeline.

**BENEFITS**

1. The HPC cloud computing environment with ANSYS 19.0 Workbench made the process of model generation really easy, with processing times reduced drastically by increasing the HPC resource.
2. The mesh models were generated for different cell numbers for moderate-fine to highly-fine mesh models. The HPC computing resource helped in achieving smoother completion of the simulation runs without re-trials or resubmission of the same simulation runs.

3. The computation requirement for a highly fine mesh (100+ million cells) is high which is near to impossible to achieve on a normal workstation. The HPC cloud provided this advantage to solve highly fine mesh models, and the simulation time drastically reduced thereby providing an advantage of getting the simulation results within an acceptable run time (about 1.5 hours).

4. The use of ANSYS Workbench helped in performing different iterations of the experiments by varying the simulation models within the workbench environment. This further helped in increasing the productivity in the simulation setup effort and thereby providing a single platform to perform end-to-end simulation model development and setup.

5. The experience with performing experiments in the HPC Cloud provided extra confidence to setup and run simulations remotely in the cloud. The different simulation setup tools required were installed in the HPC environment without any problem, and this enabled the user to access the tools without any prior installations.

6. With the use of VNC Controls in the Web browser, The HPC Cloud access was very intuitive with minimal or no installation of any pre-requisite software. The whole user experience was similar to accessing a website through the user’s own workstation browser.

7. The UberCloud containers helped with smooth execution of the project and easy access to the server resources, and provided huge advantage for the user enabling continuous monitoring of the jobs in progress without any requirement to setup the server tools in the desktop.

**RECOMMENDATIONS**

1. The Advania Data Centers HPCFLOW computing environment with HPE’s IaaS cloud management stack is an excellent fit for performing advanced computational simulation experiments that involve high technical challenges and require highly scalable hardware resources to perform the simulation.

2. There are different high-end software applications which can be used to perform Aerodynamics CFD simulation. The ANSYS 19.0 Workbench environment helped us to solve this problem with minimal effort in setting up the model and performing the simulations.

3. The combination of HPE/Advania/UberCloud/ANSYS helped in speeding up this simulation project remarkably and also completed the project within the stipulated time frame.

---

*Case Study Author – Praveen Bhat*
Technical Computing in the Advania Data Centers Cloud

Aerodynamic and CFD Simulations using Advania Data Centers’ HPCFLOW Technology

MEET THE TEAM
End-User/CFD Expert: Andre Zimmer, Managing Director, MantiumCAE
Resource Provider: Aegir Magnusson, Staffan Hansson, Hans Rickardt, Elizabeth Sargent, Advania Data Centers

ABOUT MANTIUMCAE
Based in Germany, MantiumCAE is an engineering consulting firm dedicated to computational fluid dynamics (CFD) simulations, with a particular focus on aerodynamics, optimization and CFD process automation. They assist manufacturing clients in establishing, enhancing, and optimizing their CFD capabilities and work to create products with greater aerodynamic performance.

As a specialized computer-aided engineering (CAE) consultant, MantiumCAE experiences both large and fluctuating computational demands to work on challenging projects. While browsing for on-demand HPC providers on Cloud 28+, MantiumCAE discovered Advania Data Centers (ADC) and learned about their HPCFLOW service. MantiumCAE reached out to ADC’s HPC experts and consulted with them, and subsequently determined that the best approach was to execute a hybrid approach to cloud-based HPC. This allowed them to combine their existing in-house HPC infrastructure with on-demand HPC resources from ADC. The result is a flexible approach which allowed MantiumCAE to make the most out of its existing HPC investments while increasing its ability to scale up HPC resources quickly and efficiently for its customers.

USE CASE
This case study shows how ADC’s HPCFLOW computing resources allowed MantiumCAE to create a CFD simulation quickly and efficiently for the Silvermine 11SR sportscar. To achieve this, MantiumCAE set up a CAE computing environment using UberCloud application software containers which provide a direct gateway to Advania’s HPCFLOW environment where simulations could be carried out quickly and efficiently.

“After logging into the Advania Data Centers system, running a CFD case created by MantiumFlow is just a matter of starting it. This makes engineers’ life simple."
A typical external vehicle aerodynamics simulation needs between 2,000 and 10,000 CPU core hours to be processed. Processing this simulation would take weeks to run on a 16-core workstation, but by using the HPCFLOW cloud environment together with MantiumFlow, MantiumCAE is able to deliver results within one business day.

**METHOD**

In order to successfully create and carry out the CFD simulations for the Silvermine 11SR, MantiumCAE needed the following:

- CFD Engineer with a workstation
- MantiumFlow for the CFD setup
- HPC computing power from ADC
- MantiumFlow for post-processing

The process for running the CFD simulation using HPCFLOW is very straightforward. First, the engineer creates the CFD case using MantiumFlow, which automates the setup process and uploads it to ADC’s HPCFLOW environment.

The engineer then runs the CFD simulations with a script created by MantiumFlow on the ADC environment.
Afterwards a report containing a series of plots and images is automatically created by MantiumFlow. The almost fully automated approach minimizes user error and ensures that simulations can be repeated. Everything is executed using a desktop-like environment which is easy to use and navigate.

BUSINESS BENEFITS AND NEXT STEPS
By successfully using ADC’s HPCFLOW technology, MantiumCAE was able to execute HPC CAE projects on a scale that was previously unattainable, and with a flexibility that allowed them to serve their clients’ needs better and faster. This was done without any upfront investment in computers or facilities. MantiumCAE benefitted greatly from the flexibility of the HPCFLOW service, which allowed it to scale its use of HPC resources up and down to meet its changing demands and pay only for what was needed. ADC’s HPC nodes proved to be well-suited to CFD (RAM per CPU core) and were able to process workloads quickly and efficiently.

By giving MantiumCAE access to a dedicated HPC engineer for technical support throughout the project process, ADC ensured that there was always someone available to answer questions or troubleshoot problems. They listened to MantiumCAE’s needs and provided an excellent level of service and support. This, combined with ADC’s low cost per hour, made the experience very positive.

As a result of its work with Advania Data Centers, MantiumCAE’s has greatly strengthened its ability to compete more competitively for challenging projects without high initial investments and high cost of on-demand resources. This has secured their existing business, opened new markets and positioned them well for future growth.

Case Study Authors – Andre Zimmer and Elizabeth Sargent
Thank you for your interest in our free and voluntary UberCloud Experiment!

If you, as an end-user, would like to participate in an UberCloud Experiment to explore hands-on the end-to-end process of on-demand Technical Computing as a Service, in the Cloud, for your business then please register at: [http://www.theubercloud.com/hpc-experiment/](http://www.theubercloud.com/hpc-experiment/).

If you, as a service provider, are interested in building a SaaS solution and promoting your services on the UberCloud Marketplace then please send us a message at [https://www.theubercloud.com/help/](https://www.theubercloud.com/help/).


The UberCloud Experiments and Teams received several prestigious international Awards, among other:


If you wish to be informed about the latest developments in technical computing in the cloud, then please register at [http://www.theubercloud.com/](http://www.theubercloud.com/) and you will get our free monthly newsletter.

With Support From:

![UberCloud](https://www.theubercloud.com)

Please contact UberCloud help@theubercloud.com before distributing this material in part or in full.

© Copyright 2018 UberCloud™. is UberCloud a trademark of TheUberCloud Inc.