PRONET Update - 15

QUARTERLY REPORT OF code_aster PROFESSIONAL NETWORK

Continuity of exchanges

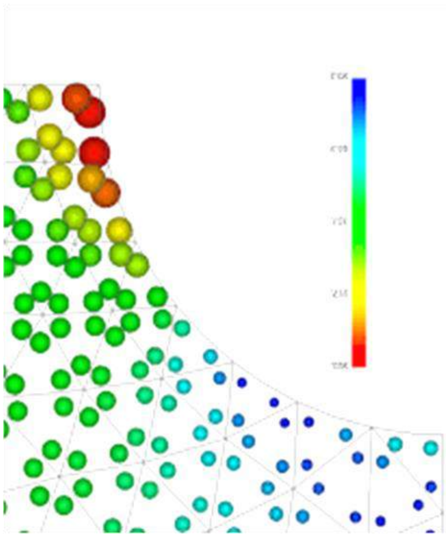
Next virtual meeting

Thursday December 3

2 p.m. to 5 p.m.

Information and registration by mail with
contact@code-aster-pronet.org

salome_meca
EXPERIENCE OPEN SIMULATION



code_aster *professional network*
user community of
code_aster and salome_meca

Fifteenth edition in English
in Spanish (SCOPE Ingenieria)
in Italian (Alter Ego Engineering)

Information content:

- Open source and ProNet
- code_aster as a research platform
- code_aster as an industrial platform
- code_aster as an educational platform
- code_aster for service providers

YouTube LinkedIn

Jean-Raymond Lévesque
Sylvie Courtier-Arnoux
Representatives of *code_aster ProNet*
contact@code-aster-pronet.org

New members Since 01 / 2020

FRANCE

SOKARIS Ingenierie

le cnam

Cairn Devices

ARGENTINA

Universidad Tecnologica Nacional

CANADA

Keovia Solutions

UK

AeroSIFT Empowering Engineers to Innovate

August 2020

Active members in 23 countries in the world

TRAINING

*Normally several **training sessions** for* **code_aster** *and* **salome_meca** *are proposed each year,* but in the context of the COVID 19 pandemic, some adaptations were necessary : **please contact the different teams directly.**

PHIMECA Solutions for robust engineering — www.phimeca.com/Formations

Alter Ego — www.aego.ai/training

DeltaCAD — www.code-aster-services.org

Dr.-Ing. J. Ackva Ingenieurbüro für Mechanik — www.code-aster.de/services

CEVAA — www.cevaa.com

Ingenieurbüro Vonstein & Partner — www.vonstein-partner.de

Technical Courses Cursos Técnicos

"***code_aster course***" (online), is available to start on 2nd of October 2020.

The code_aster course has 50 teaching hours to be finished in 8 weeks. At the end of the course an aptitude certificate will be issued to the students who pass the course.

The students receive manuals in pdf format, video-tutorials and solved exercises. An online platform is available 24 hours/7 days a week.

The teacher, **Miguel CERROLAZA** has experience with code_aster. Ph.D in Industrial Engineering by Universidad Politénica de Madrid (Spain), Master in Civil Engineering by Universidade Federal de Río de Janeiro (Brazil). Visiting profesor at Ecole Nationale des Ponts et Chaussées (France) and Colorado School of Mines (USA).

He has more than 30 years of experience in finite elements, numerical methods and computer modeling. He has written almost 10 books and dozens of papers in scientific journals. He has been invited to more than 40 conferences around the world.

The whole material (video-tutorials, exercises and texts) is available from the first day and there is no schedule. Video-tutorials are recorded. This enables each student to progress according to his learning pace.

The course structure is the following:

- 11 chapters about theory and solved exercises
- Obligatory exercises to pass the course
  - Optional exercises to improve the level

SCOPE INGENIERIA — www.scopeingenieria.com

Educational Challenges in using Open Source Finite Element software

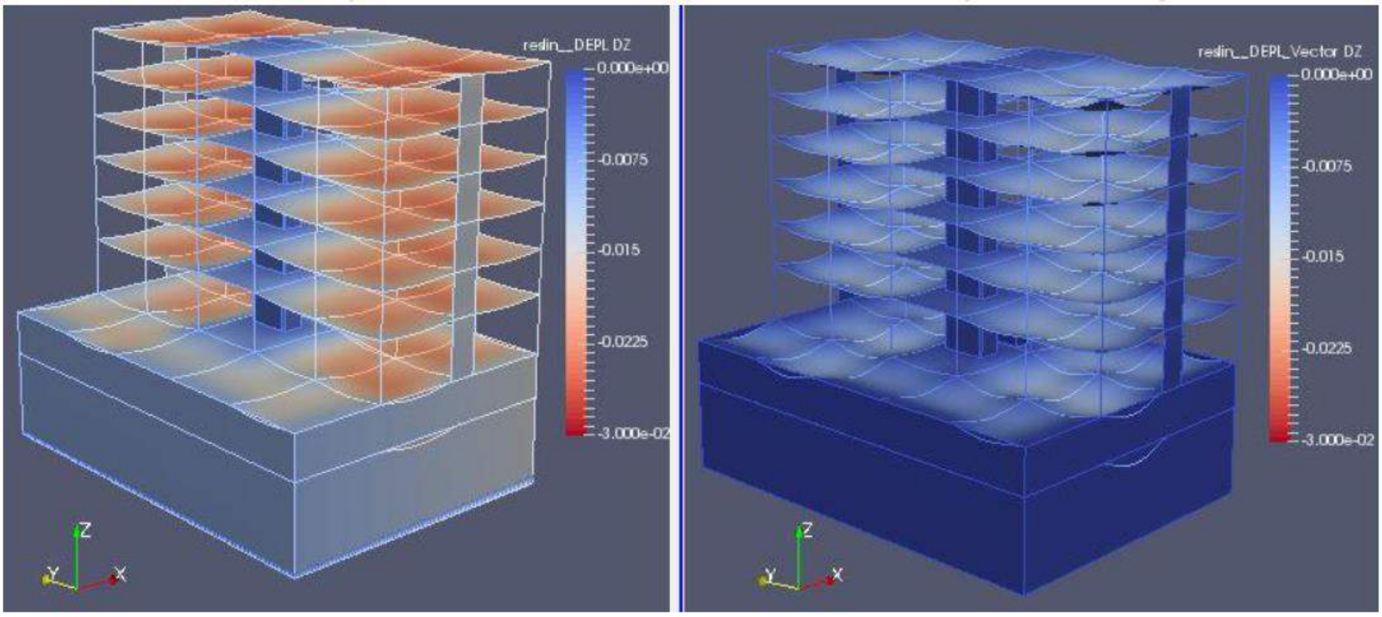
Prof. Michele BETTI

University of Florence – Dipartimento di Ingegneria Civile e Ambientale – Italy

In the Civil and Structural Engineering field, FE analysis has become an essential daily-tool commonly used to perform a broad range of tasks (e.g., damage and/or structural identification, response predictions to various loads, etc.), and the results of computer simulations constitute an important support input for decision looking for civil and structural problem solving. Due to the increasing complexity of the problems to be approached, together with the need to design new structures and infrastructures that are resilient to natural environment (e.g., wind loads, earthquakes load, fire loads, floods load, etc.), the human-skills required to master advanced FE analysis are increasingly growing.

In general, more sophisticate (or demanding) FE modelling requires more input variables to characterize the physical problem, this leading also to additional greater epistemic uncertainties. In addition, in case of existing structures or infrastructures, numerical models are usually tuned through comparison with available experimental results (the models are fitted in order to reproduce a reduced number of measurements identifying the unknown input parameters by an inverse strategy) but, at the same time, they are employed to predict the structural behaviour under exceptional loads, and predictions for new extreme load cases may be inaccurate [1].

All these aspects require that, apart the theoretical skills (e.g. numerical methods, advanced mathematics, elasticity theory, engineering sciences, computational algorithms, etc.), the Engineers involved in FE analysis must deeply know the specific software (i.e. they have the skills to judge the reliability of the numerical results). This means that it is of paramount importance that engineers have received not only a theoretical education but also a solid FE university education [2].



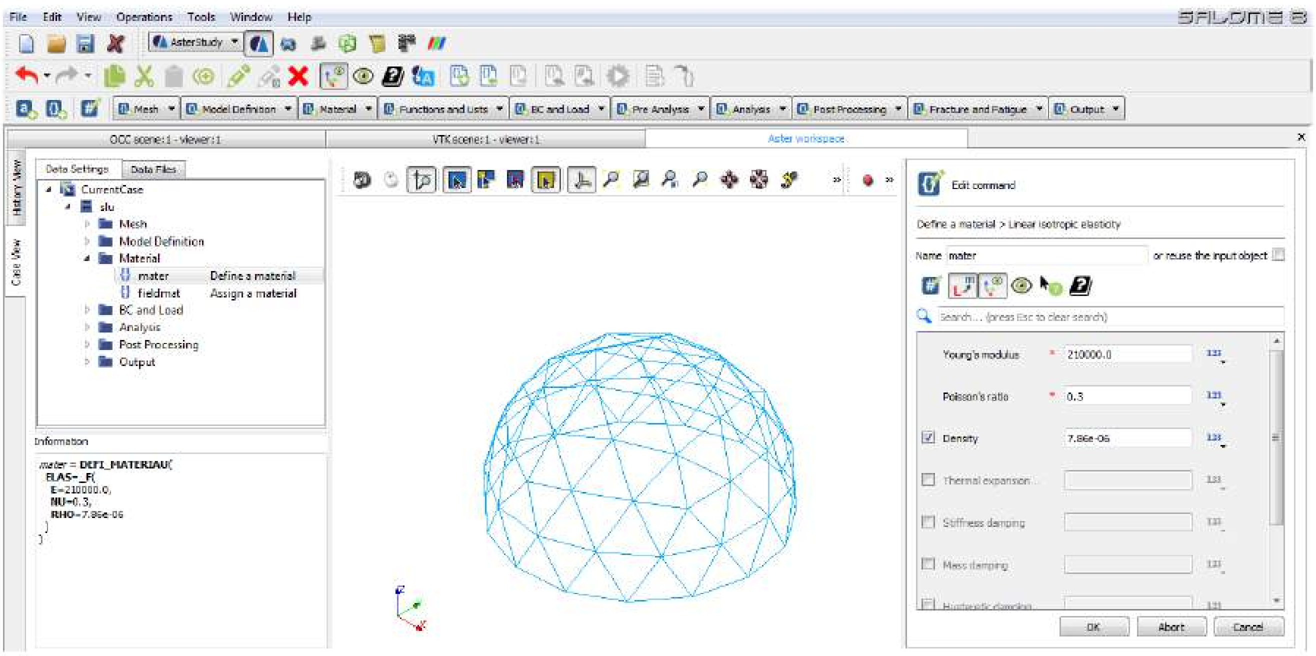
Static analysis of a RC multi-storey building (Project of F. Balloni and M. Sparacino; tutor Vladimir C. Kovačević).

Today numerous FE codes are available (both commercial and "free), and it is not unusual that many FE software houses claim that modelling using their software is easy; this can generate the false feeling that that even an inexperienced user, or a non-engineer user, can easily become a FE analyst. In reality it is not possible to use any FE package with safety without both a deeper knowledge of the inherent theory and a conscious knowledge of the used code. Lack of such knowledge can have dramatic consequences in the built environment and, as a consequence, to the society. In this respect, teaching should involve a combination of theoretical lectures on FE method, tutorials and a compulsory projects assignment avoiding the danger of using only "black box" FEA [2, 3].

[1] Bartoli G. et al. (2017). **Epistemic uncertainties in structural modelling: a blind benchmark for seismic assessment of slender masonry towers**. ASCE's Journal of Performance of Constructed Facilities; 31(5): 04017067. DOI 10.1061/(ASCE)CF.1943-5509.0001049.

[2] Betti M. et al. (2015). **Engineer Education and Research With Code Aster**. Proceedings of WEEF (World Engineering Education Forum. Engineering Education for a Resilient Society), Firenze, Italy, 20-24 September 2015.

[3] Plevris V., Markeset G. (2018). **Educational Challenges in Computer-based Finite Element Analysis and Design of structures**. Journal of Computer Science; 14 (10): 1351.1362. DOI: 10.3844/jcssp.2018.1351.1362.



Linear buckling of a geodetic done (Project assignment of Maria Antonietta Viciconte; tutor Vladimir C. Kovačević).

The Structural Engineering section of the Department of Civil and Environmental Engineering (UNIFI) has started, in the last years, to employ code_aster as a foundation to teach FE software engineering (and to analyse engineering problems in the field of seismic assessment of historic masonry buildings and artefacts [4]). The code is also currently employed by Bachelor and Master student to perform the numerical analyses needed to develop their thesis.

[4] Pieraccini M. et al. (2017). **Radar detection of pedestrian-induced vibrations of Michelangelo's David**. Plos ONE; 12(4): e0174480. DOI: 10.1371/journal.pone.0174480.

The goal of proposing and using the Open Source software are: to provide students with a real-world software engineering experience (that can be freely employed after their study); to introduce students to the Open Source developmental model; to attract a wider variety of students into computing due to the real-world and the ethical nature of the project; to teach a conscious and safe use of FE software.

This has been possible tanks to the support of KOBE Innovation Engineering (www.kobe-ie.com) who is also promoter of the web page Code Aster Italia (code-aster.it). The page has a tutorial dedicated page and recently, thanks to Alessandro Antonelli, some of the code_aster documents translated in Italian are reported.

The second edition of the event dedicated to Italian code_aster users was held this February 14 at the headquarters of Confindustria Firenze, the industrial footprint marks a small step towards bringing users and computer code experts closer to local companies and leads to a comparison on open source issues in innovative industrial processes.

It was the last physical meeting in the world this year before the COVID event.

It is with pleasure that we announce that the new code_aster documents translated from French to Italian by Alessandro Antonelli.

Some of U2 section new documents are: U2.04.01 on the STAT_NON_LINE solver instruction and U2.04.04 on the contact instruction.

In the U4 section you can view the new U4.72.04 CREA CHAMP document and the new U4.91.04 IMPR CONCEPT document.

A new Concept of Learning code-aster with Video Seminar

Johannes ACKVA
Ingenieurbüro für Mechanik – Germany

As Ingenieurbüro für Mechanik (www.code-aster.de) we are offering since more than 10 years Code-Aster "in-class" seminars (with physical presence) in Germany to industrial clients. These courses have continuously been updated (to actual versions of the software) and further developed over the years with the feedback of participants and of support clients, and also with stimuli taken from numerous questions in the code_aster forums.

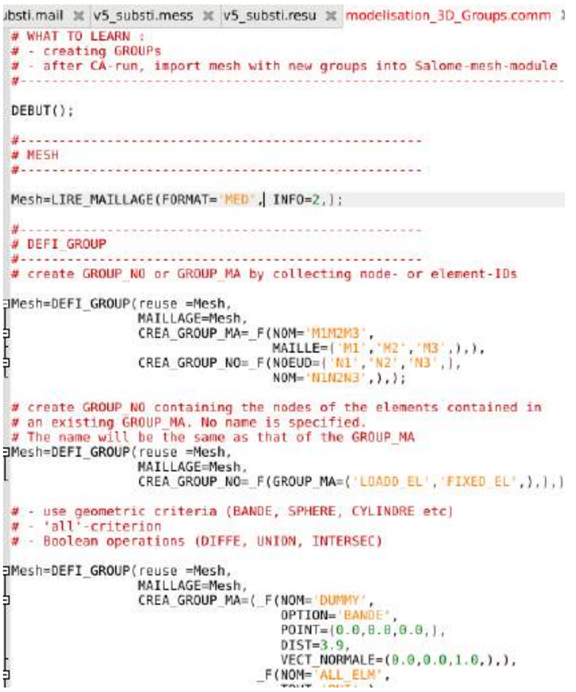
Less than half of the participants came from Germany, the others being from European countries, most of them from Austria and Switzerland. Requests from more distant countries could also be satisfied in numerous cases through Inhouse seminars, the most distant at Professor Torsten Calvi Corporation, Manila / Philippines (www.ptcc.design).

Due to the large distances to business partners and also considering the situation of the Covid virus we are now setting up Video courses. Participants can learn without moving from their company or from their homes and at their own pack.

The new professional video courses have been developed also with the consultation of experts for online-Learning from the University of Applied Sciences, Munich, Professor Dr. Dirk Burth et al.

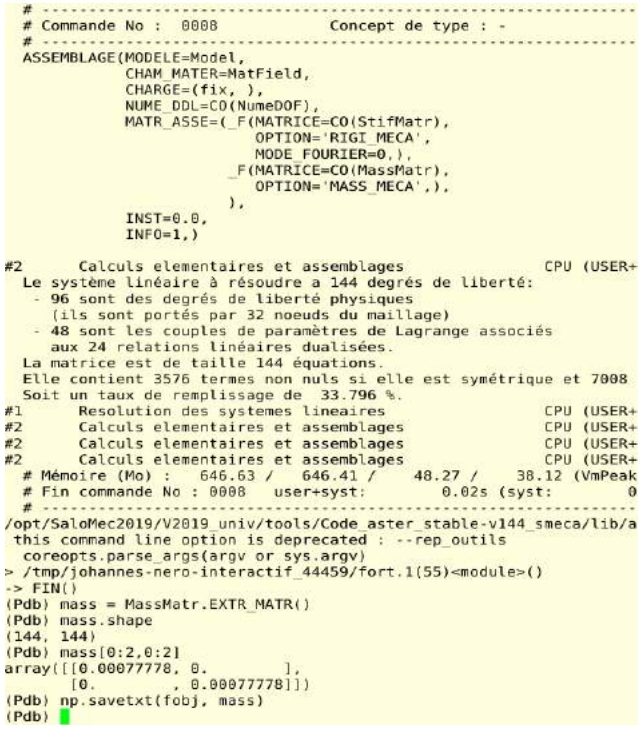
The video courses composed of 3 parts will be available from about end of October on our website www.code-aster.de:

(1) the complete course folders with all the Code-Aster examples and case studies which are the nucleus of our well-proven in-class courses, for example of our 5 days Introduction Course
(2) videos explaining these folders and helping in acquiring the content
(3) web conferences as needed



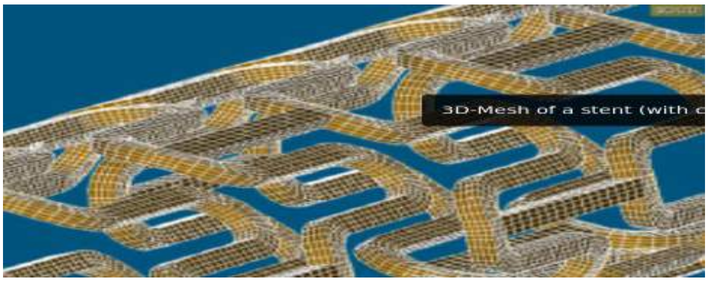
```
ıbsti.mail  v5_substi.mess  v5_substi.resu  modelisation_3D_Groups.comm
# WHAT TO LEARN :
# - creating GROUPs
# - after CA-run, import mesh with new groups into Salome-mesh-module
#------------------------------------------------------------------------

DEBUT();

#---------------------------------------------------
# MESH
#---------------------------------------------------

Mesh=LIRE_MAILLAGE(FORMAT='MED',INFO=2,);

#---------------------------------------------------
# DEFI_GROUP
#---------------------------------------------------
# create GROUP_NO or GROUP_MA by collecting node- or element-IDs

Mesh=DEFI_GROUP(reuse =Mesh,
                MAILLAGE=Mesh,
                CREA_GROUP_MA=_F(NOM='M1M2M3',
                                 MAILLE=('M1','M2','M3',),),
                CREA_GROUP_NO=_F(NOEUD=('N1','N2','N3',),
                                 NOM='N1N2N3',),);

# create GROUP_NO containing the nodes of the elements contained in
# an existing GROUP_MA. No name is specified.
# The name will be the same as that of the GROUP_MA
Mesh=DEFI_GROUP(reuse =Mesh,
                MAILLAGE=Mesh,
                CREA_GROUP_NO=_F(GROUP_MA=('LOADD_EL','FIXED_EL',),),)

# - use geometric criteria (BANDE, SPHERE, CYLINDRE etc)
# - 'all'-criterion
# - Boolean operations (DIFFE, UNION, INTERSEC)

Mesh=DEFI_GROUP(reuse =Mesh,
                MAILLAGE=Mesh,
                CREA_GROUP_MA=(_F(NOM='DUMMY',
                                  OPTION='BANDE',
                                  POINT=(0.0,8.0,0.0,),
                                  DIST=3.9,
                                  VECT_NORMALE=(0.0,0.0,1.0,),),
                               _F(NOM='ALL_ELM',
```

Mastering the command DEFI_GROUP can considerably reduce the work with Salome, creating the groups in an automatic manner



```
# ------------------------------------------------------------------------
# Commande No :  0008            Concept de type : -
# ------------------------------------------------------------------------
ASSEMBLAGE(MODELE=Model,
           CHAM_MATER=MatField,
           CHARGE=(fix, ),
           NUME_DDL=CO(NumeDOF),
           MATR_ASSE=(_F(MATRICE=CO(StifMatr),
                         OPTION='RIGI_MECA',
                         MODE_FOURIER=0,),
                      _F(MATRICE=CO(MassMatr),
                         OPTION='MASS_MECA',),
                      ),
           INST=0.0,
           INFO=1,)

#2      Calculs elementaires et assemblages                    CPU (USER+
  Le système linéaire à résoudre a 144 degrés de liberté:
   - 96 sont des degrés de liberté physiques
     (ils sont portés par 32 noeuds du maillage)
   - 48 sont les couples de paramètres de Lagrange associés
     aux 24 relations linéaires dualisées.
  La matrice est de taille 144 équations.
  Elle contient 3576 termes non nuls si elle est symétrique et 7008
  Soit un taux de remplissage de  33.796 %.
#1      Resolution des systemes lineaires                      CPU (USER+
#2      Calculs elementaires et assemblages                    CPU (USER+
#2      Calculs elementaires et assemblages                    CPU (USER+
#2      Calculs elementaires et assemblages                    CPU (USER+
  # Mémoire (Mo) :   646.63 /   646.41 /    48.27 /    38.12 (VmPeak
  # Fin commande No : 0008   user+syst:        0.02s (syst:        0
  # ------------------------------------------------------------------------
/opt/SaloMec2019/V2019_univ/tools/Code_aster_stable-v144_smeca/lib/a
 this command line option is deprecated : --rep_outils
  coreopts.parse_args(argv or sys.argv)
> /tmp/johannes-nero-interactif_44459/fort.1(55)<module>()
-> FIN()
(Pdb) mass = MassMatr.EXTR_MATR()
(Pdb) mass.shape
(144, 144)
(Pdb) mass[0:2,0:2]
array([[0.00077778, 0.        ],
       [0.        , 0.00077778]])
(Pdb) np.savetxt(fobj, mass)
(Pdb)
```

Python is useful for debugging Code-Aster commands and can be used to extract data, here for example a matrices from Code-Aster data structures



3D-Mesh of a stent

(with courtesy to Feops, Belgium

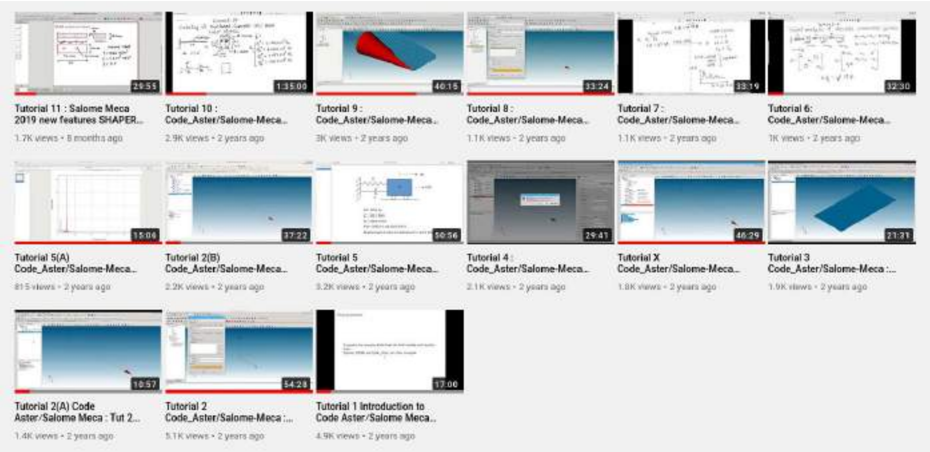
Teaching and learning code_aster go hand in hand

Digvijay PATANKAR - Senior Research Fellow

Indian Institute of Technology ROORKE – India

I discovered code_aster for the first-time way back in 2009. That was the time when I was trying to figure out an open source finite element simulation tool for my master's thesis. Among some other popular choices, code_aster stood out because of its generic interface and large library of finite elements and materials. But the learning curve was very steep to say the least (at least for a non-French speaker). Back then, the documentation was available only in french and the online translation had many flaws. This made learning it more difficult. Nevertheless, I continued learning and soon realized that once the initial hurdle is crossed, it is much easier to follow.

One major source for learning was soon available – a book by Jean Pierre Aubry. I am sure that book has helped many (including me) to understand the structure of code_aster command files. Yet, as a medium, book has its own limitations. Moreover, the somewhat rigid separation of preprocessor, postprocessor and solver was also a major issue for an absolute beginner. Although, this was partly addressed in SALOME-MECA platform, the integration of various components wasn't very seamless.



This changed rapidly in 2017 when a module « AsterStudy » was introduced into the SALOME-MECA platform and this made the things a lot easier for beginners.

This is when I thought that I can try to make the beginners entry into code_aster easier by making screencast videos of finite element modeling using SALOME-MECA.

The main aim was initially to introduce the software, the basic workflow, integration of various components and how to handle them effectively. At that time there were not many videos on code_aster. As a result, the videos gained enough attention and were used by many to learn.

I have received many emails congratulating/thanking me for the videos. I must mention here that all the videos are very basic and cover only the linear elastic mechanical analysis. This shows that, the beginner is usually not interested in exploring all the features but rather wants to find them out on their own but needs a hand holding in the beginning phase where (s)he wishes to explore a new software.

My doctoral studies supervisor Prof. Manish SHRIKHANDE was planning a short-term course on « Finite Element Method and Computational Structural Dynamics ». The course had hands on sessions on using FE simulation tool. Since both of us prefer to use open source tools as daily drivers, and I was supposed to be an instructor for the hands-on sessions, code_aster was chosen as a tool. None of the participants involved were earlier exposed to code_aster. The feedback obtained from them made me rethink about the video content and are going to have a significant impact on future videos.

During this entire journey, I have learned a lot about the software. As it is often said, **« If you want to learn something, teach it to others ».**

CubeSat structural verification as a teaching method

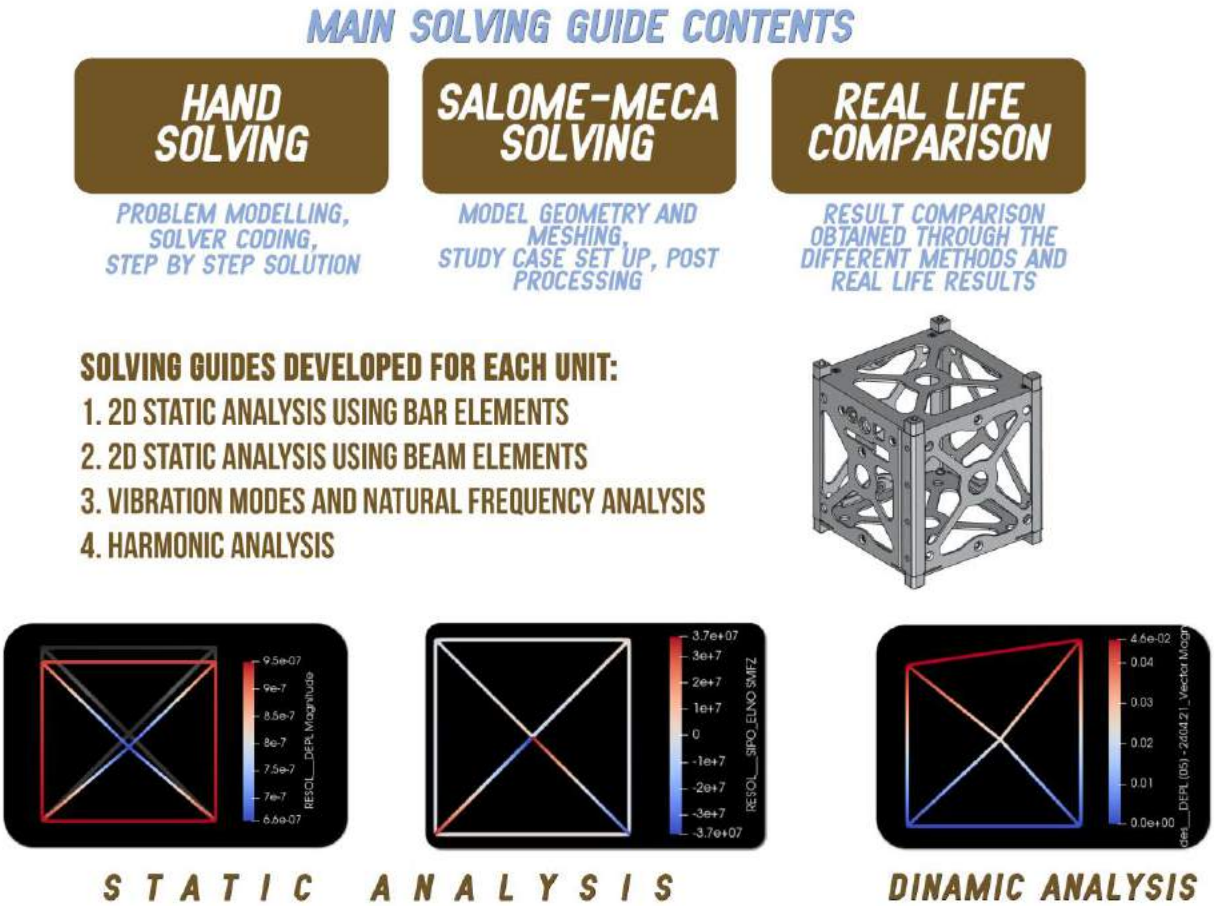
Eng. Nicolás COPPOLECCHIA – Eng. **Ignacio CAPPARELLI**

Universidad Tecnologica Nacional - Facultad Regional Haedo – Argentina

During the year 2020, at Aeronautical Structures 3 course we have adapted our examination methods due to the COVID-19 pandemic. Given this scenario we have chosen to further include the use of Code_Aster for different tasks.

Our objective was to use the different methods explained during the year to analyze the structure of an actual CubeSat model. CubeSats are one of the different types of nanosatellites, which are raising worldwide interest in universities and companies due to their low cost and simple design.

Students had to analyze, through a **solving guide** made by the professors, the behaviour of the CubeSat under different load conditions using methods seen on each unit.



According to feedback, our students are satisfied with this methodology because they can verify the accuracy of the hand solving methods as well as the software accuracy. Our main goal is satisfied, because students understood what simulation software does “behind the scenes” when solving a problem.

Solving guides are available here: https://rb.gy/mpo6vg

Real life case: Melahat Cihan, Ozan Haktanr, lke Akbulut, and A.R. Aslan. Flight dynamic analysis of itupsat1. 01 2008.

Bridging the Gap between Industry and Academia through Student Project works in Aircraft Structures Analysis

Vamsi MULA , cEng MIET

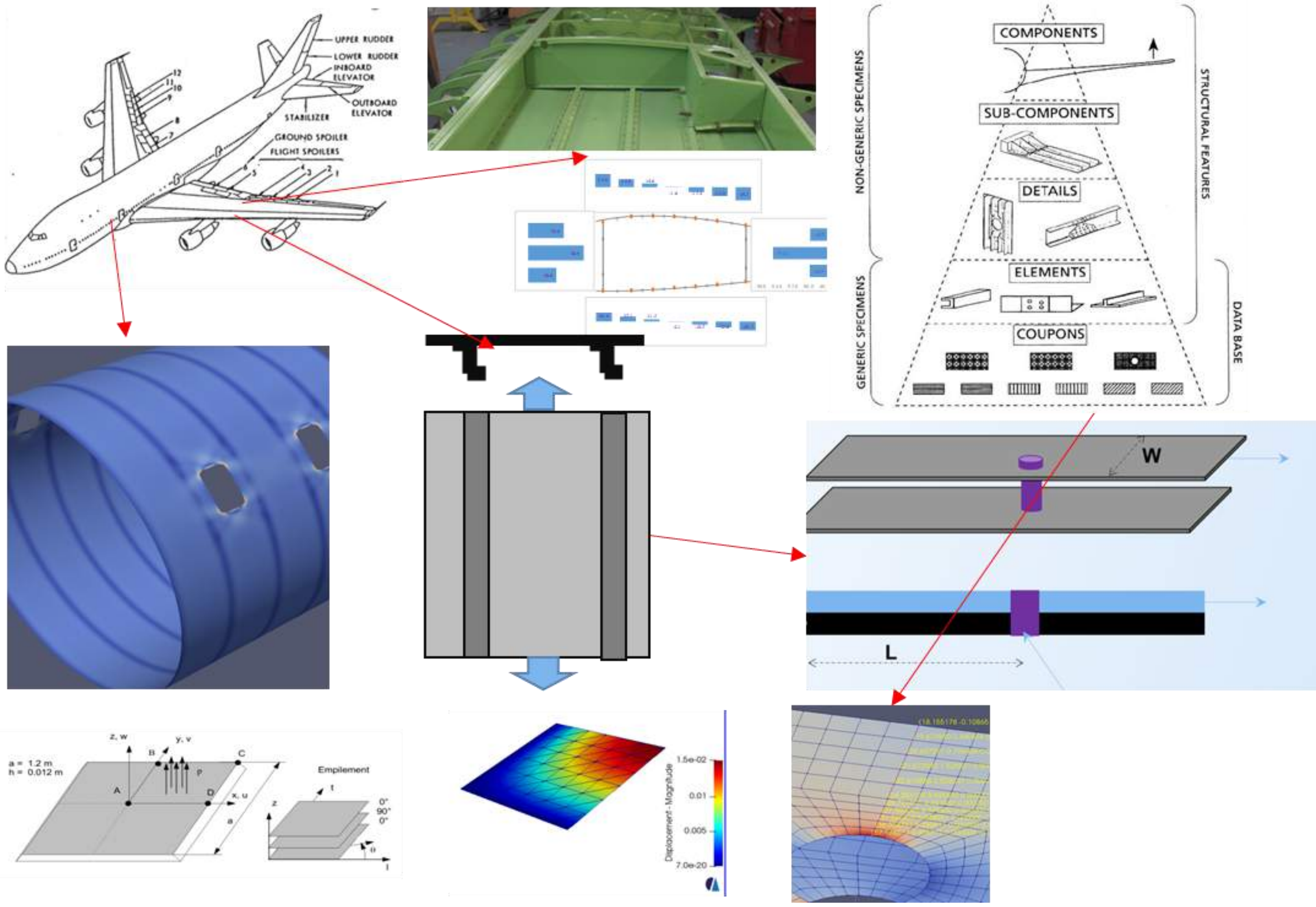
Structures Consultant AeroSIFT Limited – UK

AeroSIFT Limited provides consultancy and training services to Aircraft Structural Repairs and Modifications Design organizations. Other important working area following our passion is Bridging the Gap between Industry and Academia through student project works in the same area. Our aim is to undertake all aspects of Aircraft Structural analysis from pre design till retirement addressing the aspects of design studies, detailed design calculations, production and in-service support of Aircraft Structure.

With recent advances in Data Science and the concepts of Digital Twins to maintain the aircraft in-service, Finite Element analysis will be even more useful in efficiently maintaining the aircraft structural integrity.

We intend to build a database of tutorials covering all aspects of Aircraft Structural analysis within the Building Block approach typically used in product developments as shown below. The tutorials will show usage of Salome Meca / code_aster along with the detailed analytical calculations focusing on

- Design and Analysis of Efficient Metallic Components as well as Advanced Composite Materials
- Fracture Mechanics for Damage Tolerance Evaluation of Metals as well as composites.
- Use of Fatigue spectrum within Code Aster for Fatigue and Damage Tolerance evaluation.



Developing an open-source openBIM IFC-driven FEM analysis workflow with code_aster

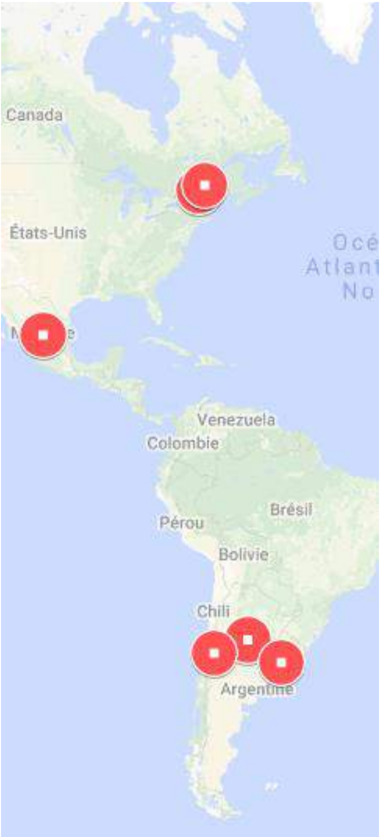
Ioannis P. CHRISTOVASILIS - Aether Engineering – Italy

Aether Engineering has been engaging in the development of an open-source workflow for structural analysis with *code_aster* based on openBIM structural IFC files. This development is in collaboration with the developers of the *IfcOpenShell* library and the *BlenderBIM* add-on for Blender.

The project called *IFC2CA* includes dedicated scripts to (i) extract all necessary information from a structural IFC file to a JSON file, (ii) create the geometry and mesh in *Salome_Meca*, and (iii) create the command file for *code_aster*. The project is ongoing and has been absorbed in the official repository of *IfcOpenShell*, while a separate repository hosts some of the example implementations. Meanwhile, in *BlenderBIM* dedicated User Interface components are developed to define the geometry and the metadata needed for the structural file.

To support this project, Aether Engineering is also developing a *Structural BIM Web Viewer*, hosted at the official website, that can be used to visualize existing structural IFC files. It is also supported by a backend service to directly obtain and download the respective JSON file. The viewer is accessible from all devices including tablets and cell phones.

USEFUL LINKS:
IfcOpenShell repository: https://github.com/IfcOpenShell/IfcOpenShell
ifc2ca repository: https://github.com/Jesusbill/ifc2ca
BlenderBIM: https://blenderbim.org/
Structural BIM Viewer: https://www.aethereng.com/#/ifc_structural_viewer


Canada
États-Unis
Venezuela
Colombie
Brésil
Pérou
Bolivie
Chili


Russie
Kazakhstan
Mongolie
Chine
Afghanistan
Inde
Thaïlande
Indonésie
Océan
Indien


Suède
Finlande
Norvège
Royaume-Uni
Irlande
Biélorussie
Pologne
Ukraine
Roumanie
Grèce
Turquie
Portugal
Syrie
Mer
Méditerranée
Maroc
Algérie
Libye
Égypte