Waltham Users Forum:
Unleashing the Power of PyAnsys

Sandeep Medikonda

September 20th, 2023

Ansys

Agenda

- What is PyAnsys?
- Why use it & innovation offered?
- What's Available?
- Popular PyAnsys Workflows
  - Machine Learning Example
  - WebApps
- How to install it?
- Documentation, Help & Other Resources
- Conclusions

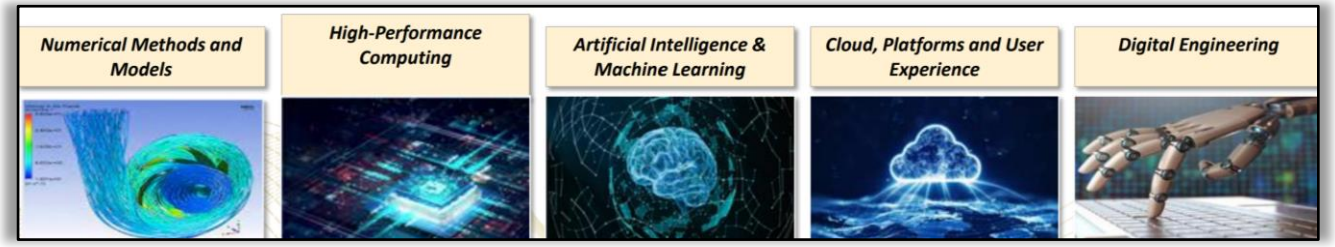
Need for APIs?

Evolving Ansys Customer Needs:

- Democratization of simulation (including non-engineers)
- Accelerate predictable insights
- Connect Ansys technologies to wider enabling technologies such as AI/ML

Ansys Assessment:

- Requirements can be classified into 5 Pillars: **Numerical Solvers**, **High-Performance Computations**, **AI/ML**, **Cloud** and **Digital** Engineering.
- Need for strong and open APIs



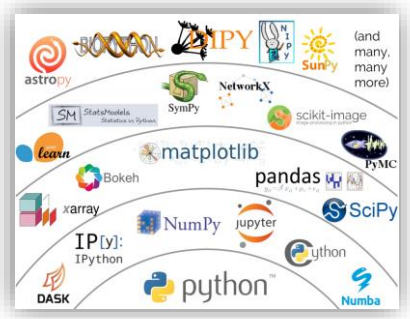
5 Pillars of Ansys Technology Strategy

Enabled by **Open**, **Clear**, **Concise**, **Maintainable** and **Remotable APIs**.

It’s all about increasing the value of our tools for customers…

What is PyAnsys?

Set of technologies that allow the user to interface with Ansys products pythonically

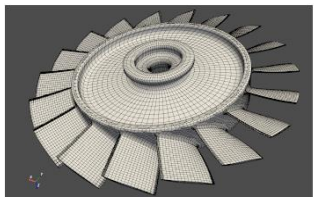


PyAnsys

Python + Ansys = PyAnsys

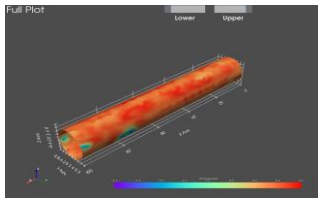
PyMAPDL

Access MAPDL simulation and solver



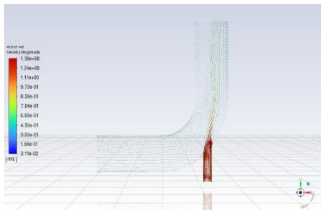
PyAEDT

Access Scripting functionalities of the AEDT Suite



PyFluent

Pythonic interface to Ansys Fluent



More…

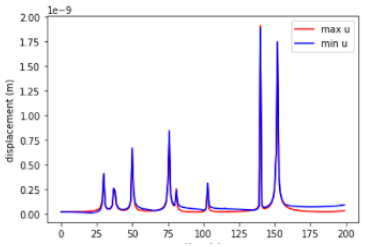
Visit… docs.pyansys.com

Post-processing librairies

DPF-Core

Read and manipulate finite element data with powerful, scalable operators

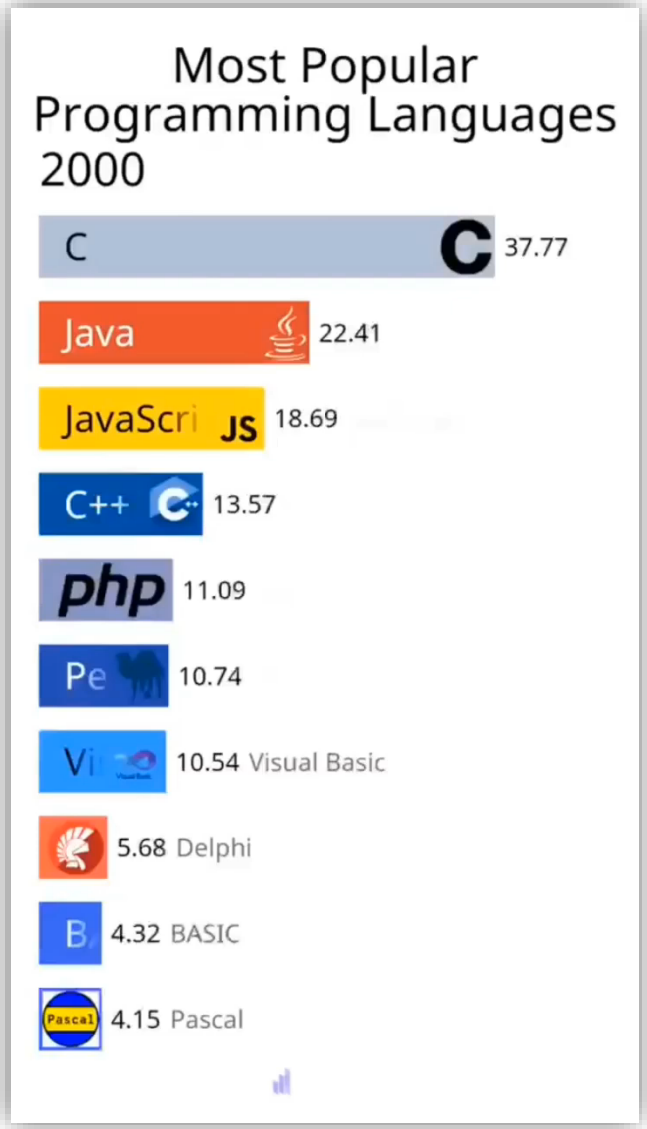
DPF-Post



Extract simulation data with a streamlined post-processing API

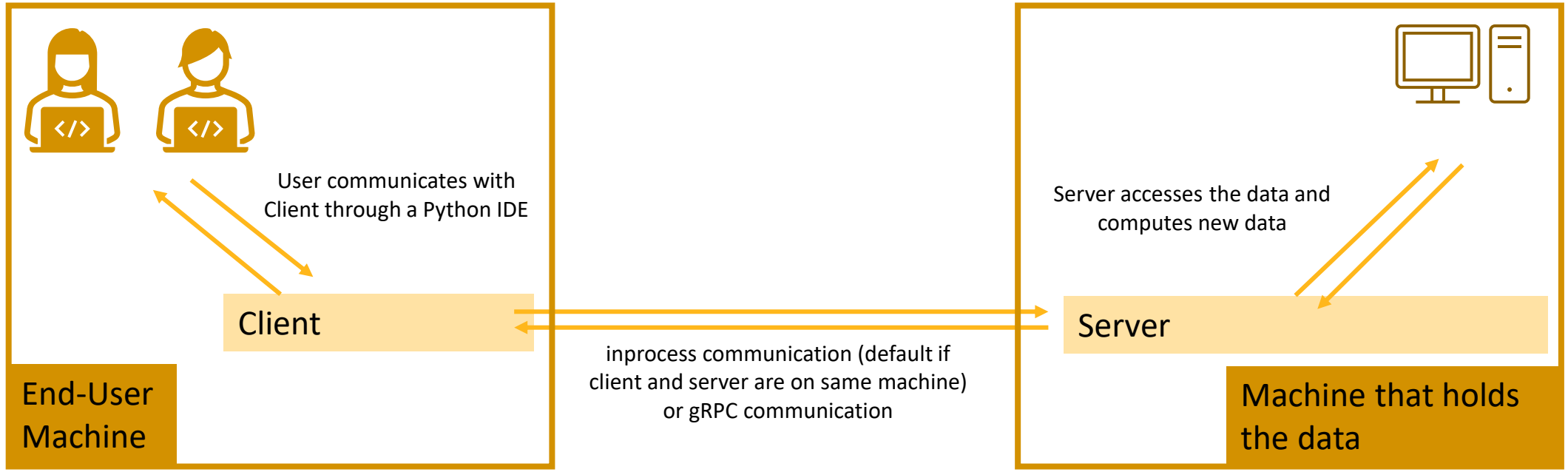
Putting the Py in PyAnsys

- Py is short for python, the language, not the snake
  - nor pie which is a shame.
- Python is becoming the *lingua Franca* of engineering software languages especially data investigation (e.g AI/ML).



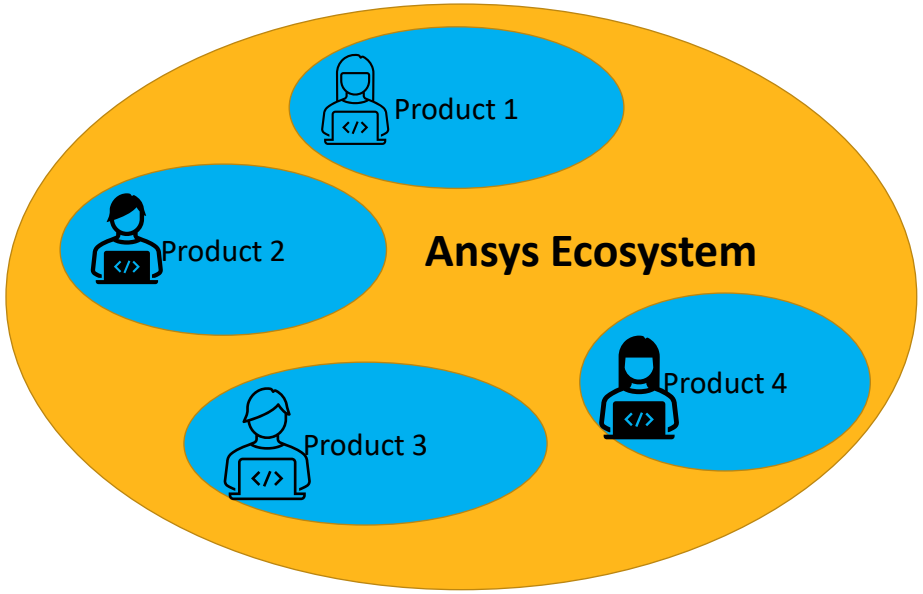
Client/Server architecture

- PyAnsys project offers several packages that are usually based on a Client/Server architecture. Client and Server can be on the same machine, or on separate machines. Below is a typical configuration:

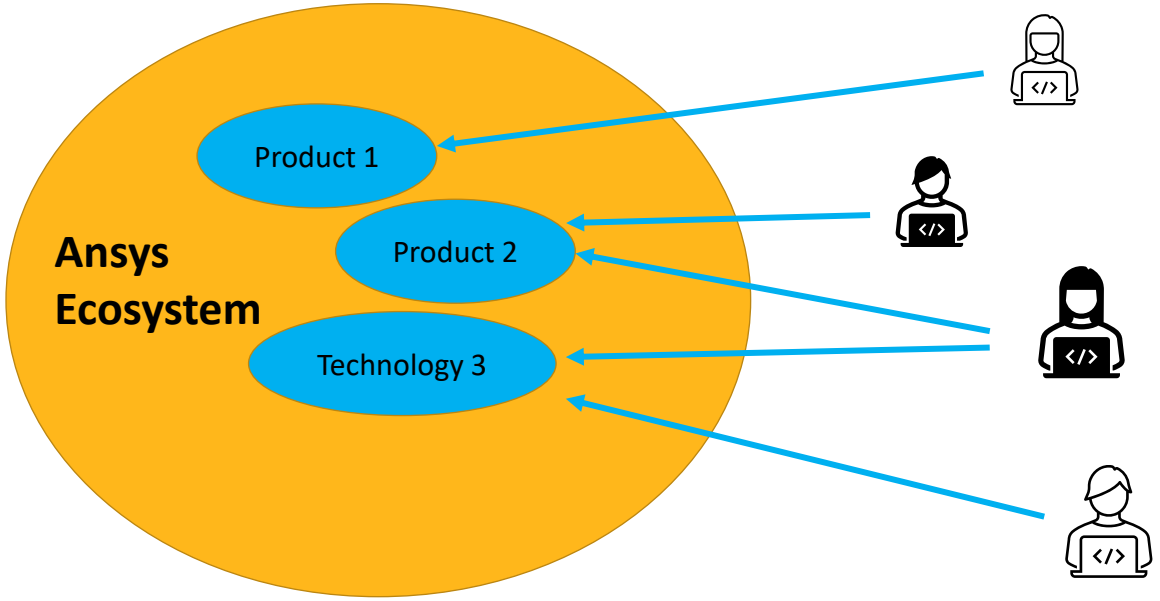


- Where and how to find the files?
  - The Client part comes is provided through Python packages (pip install)
  - The Server part is shipped with the standard Ansys installation

Two possible approaches to scripting



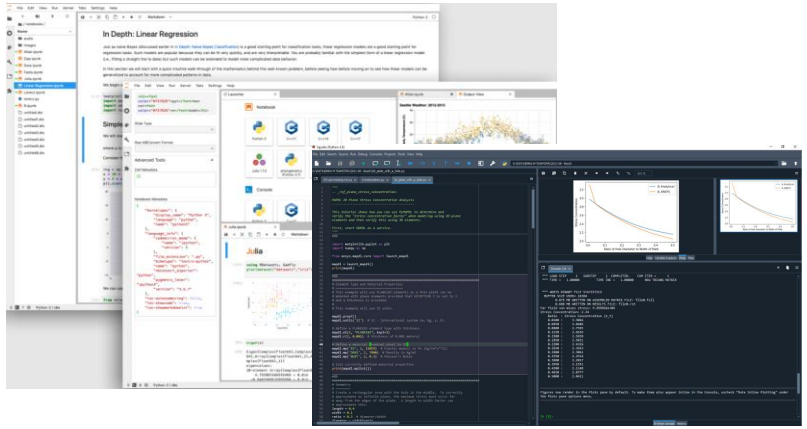
Product scripting



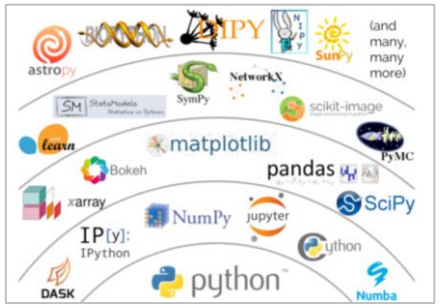
PyAnsys scripting

Scripting from within the customer tools – PyAnsys

Any tool from which you can write Python (UI is yours to choose)



Possibility to use other useful Python tools



Customer tools

PyAnsys technology

Products

- API — Ansys Product A
- API — Ansys Product B
- API — Ansys MAPDL

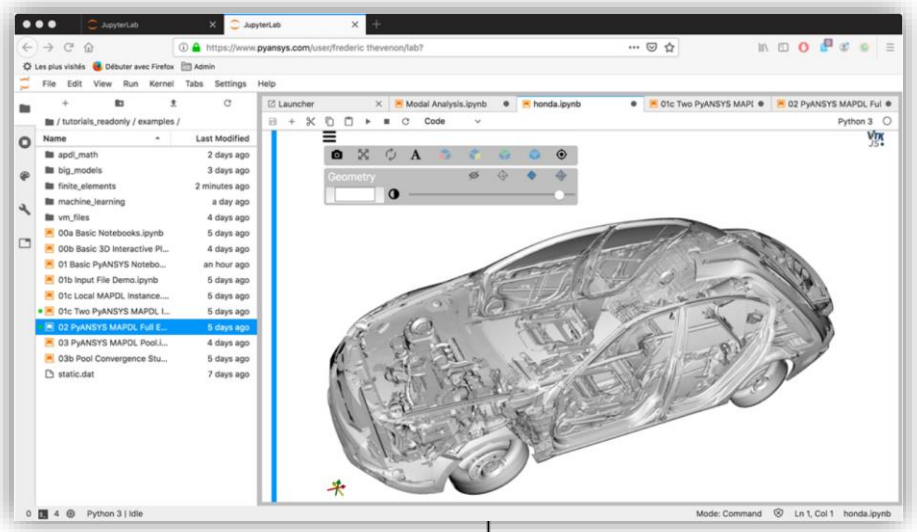
Technology components

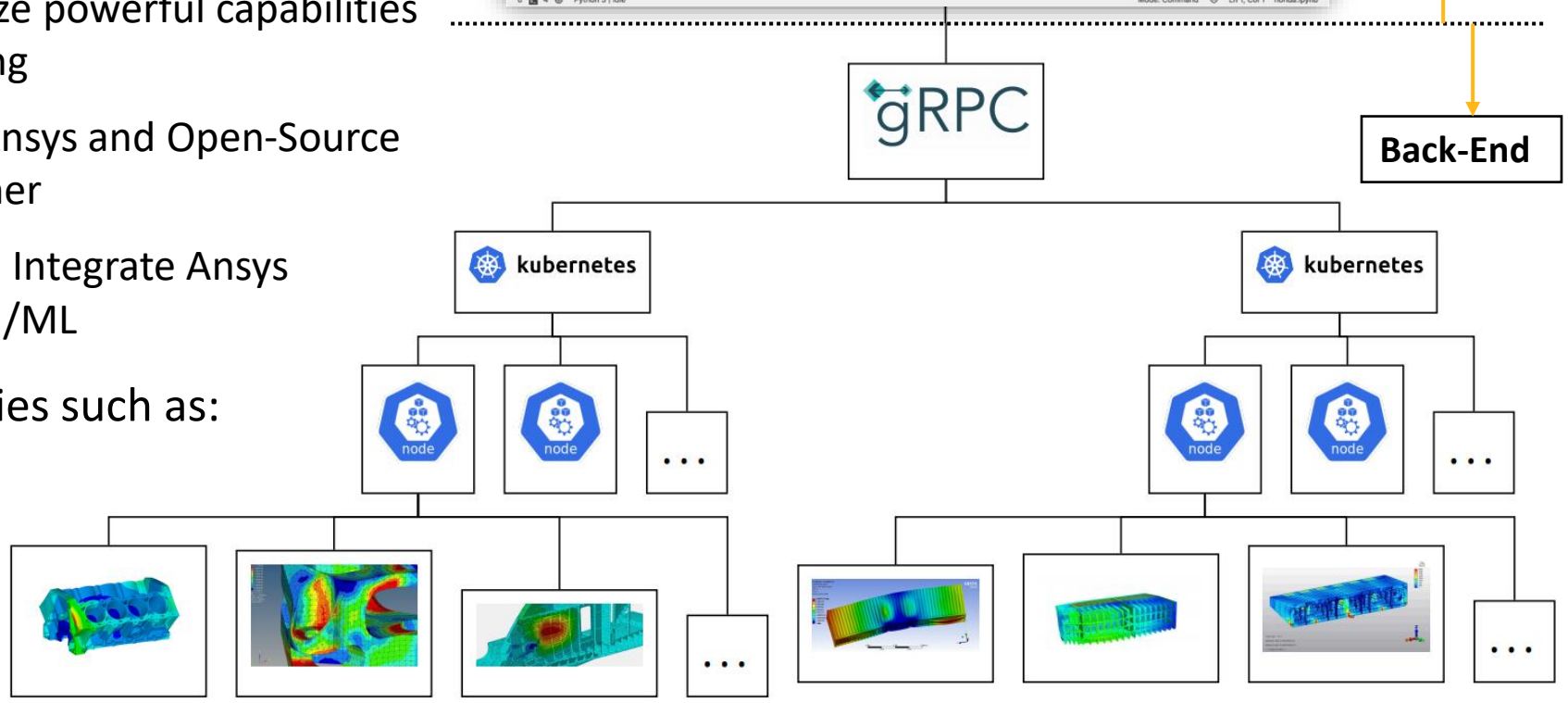
- Meshing
- Data Processing Framework

Ansys Technology

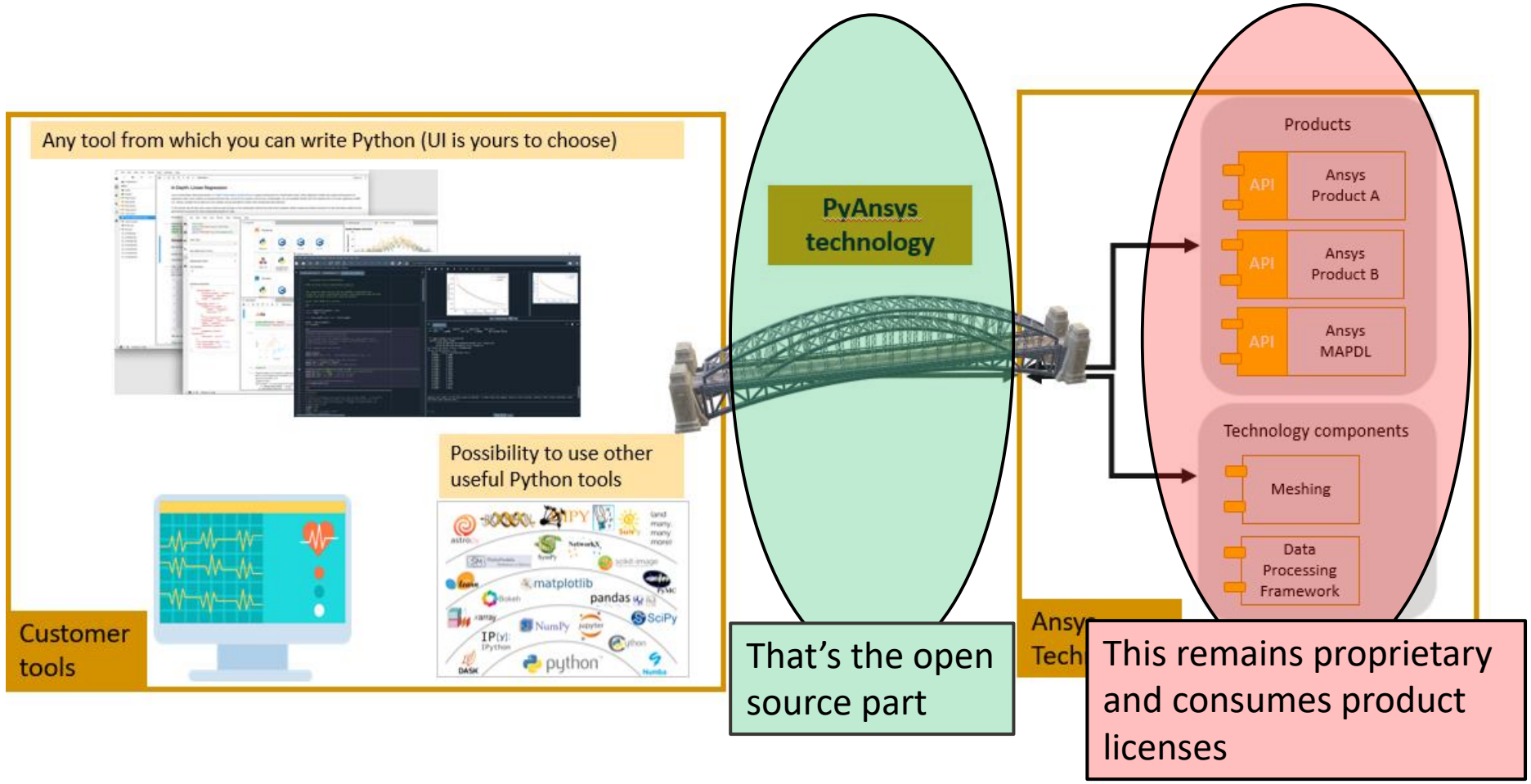
Why use it & Innovation offered?

- The PyAnsys project is **Ansys's commitment to open-source** where we provide Python libraries that expose Ansys technologies in the Python ecosystem through APIs and interfaces that are **clear**, **concise**, and **maintainable**. This allows Ansys Customers to do:
  - **Flexible Automation:** Democratize powerful capabilities offered by Ansys through scripting
  - **Flexible Distribution :** Connect Ansys and Open-Source technologies in a seamless manner
  - **Broader Technology Integration:** Integrate Ansys physics capabilities easily with AI/ML
- Built on cutting edge technologies such as:
  - gRPC
  - Kubernetes Framework (or any HPC)
  - Headless Docker Containers





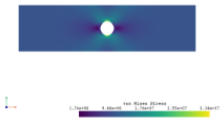
Open source ?! What's in it for us?



We're just offering a new way of using our products!
This will allow existing users to use our products differently, effectively and efficiently.

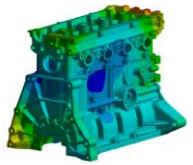
PyAnsys: What's available?

Simulation libraries



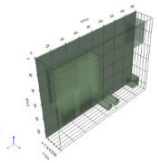
PyMAPDL

Pythonic interface to Ansys MAPDL (Mechanical APLD)



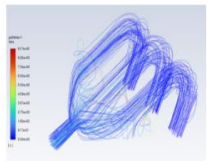
PyMechanical

Pythonic interface to Ansys Mechanical



PyAEDT

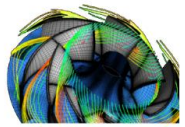
Pythonic interface to AEDT (Ansys Electronic Desktop)



PyFluent

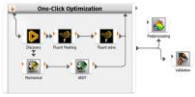
Pythonic interface to Ansys Fluent

Utility libraries



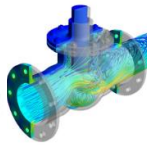
PyPrimeMesh

Pythonic interface to Ansys Prime Server, which delivers core Ansys meshing technology



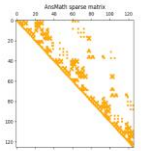
PyOptislang

Pythonic interface to Ansys Optislang



PySystem Coupling

Pythonic interface to communicate with Ansys System Coupling



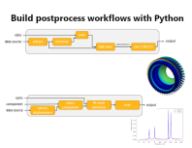
PyAnsys Math

Pythonic interface to PyAnsys Math libraries

PyDynamicReporting

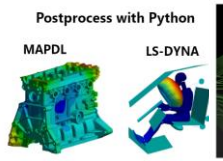
Pythonic interface to Ansys Dynamic Reporting for service and control of its database and reports

Post-processing libraries



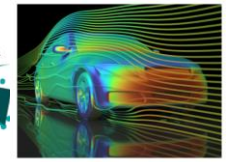
PyDPF - Core

Pythonic interface to DPF (Data Processing Framework) for building more advanced and customized workflows



PyDPF - Post

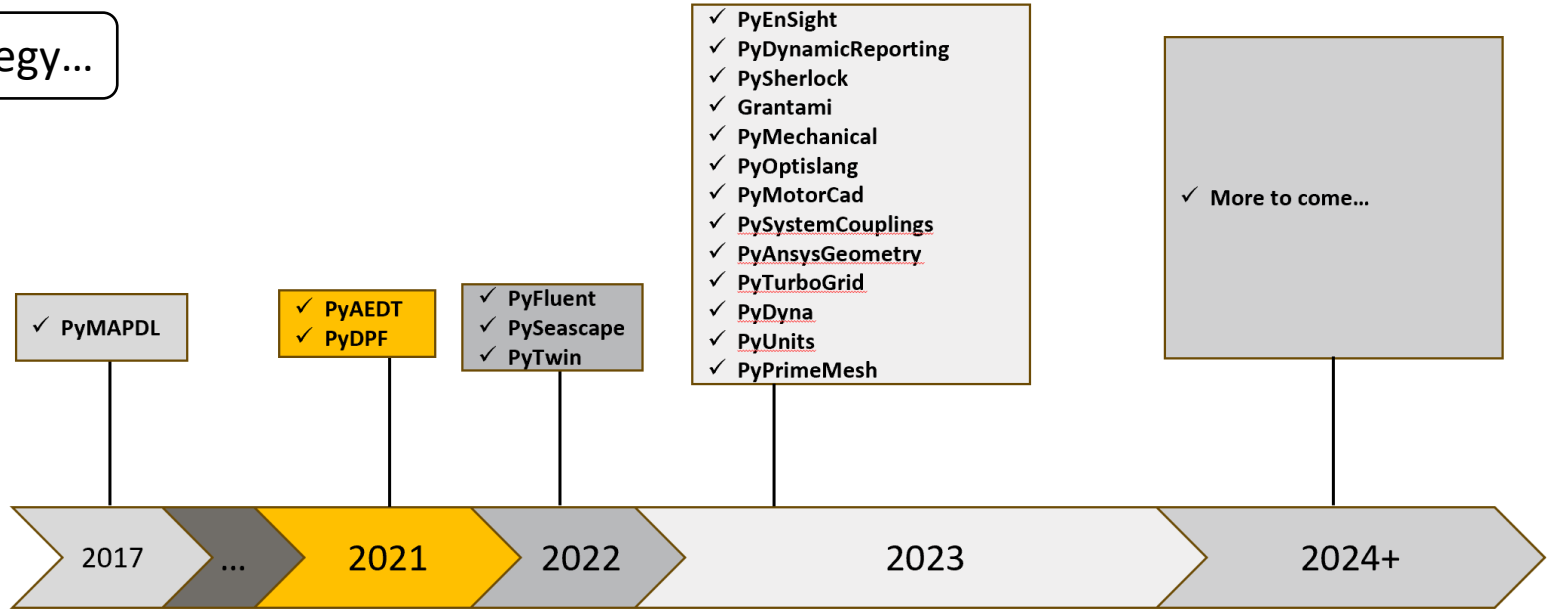
Pythonic interface to access and post process Ansys solver result files



PyEnSight

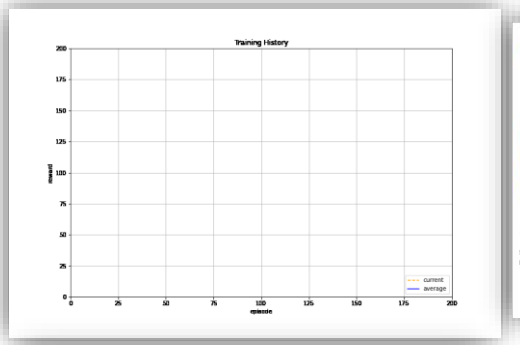
Pythonic interface to EnSight, the Ansys simulation postprocessor

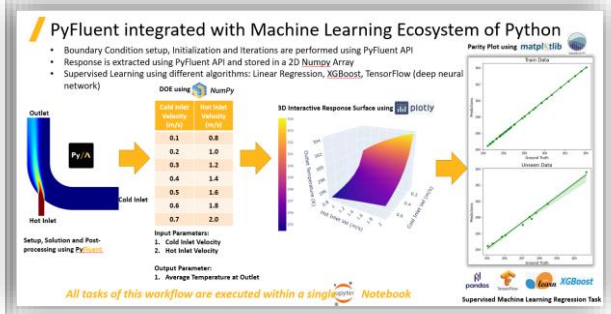
5-year strategy…

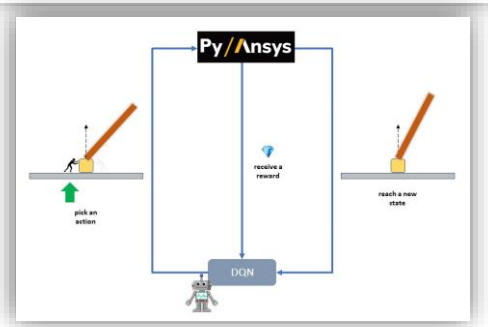


Keep an eye on…
docs.pyansys.com

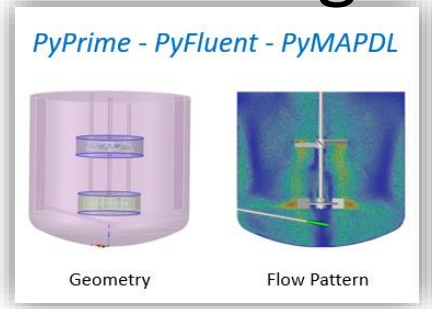
Key workflows where Ansys customers are using PyAnsys

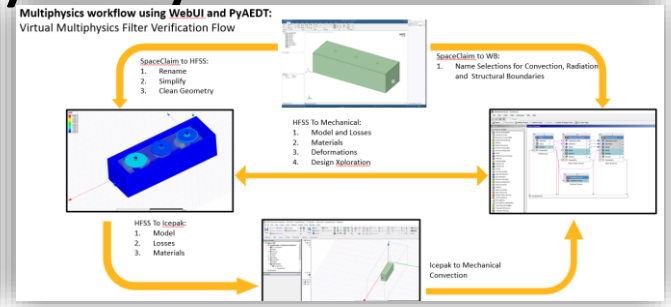






Machine Learning Problems

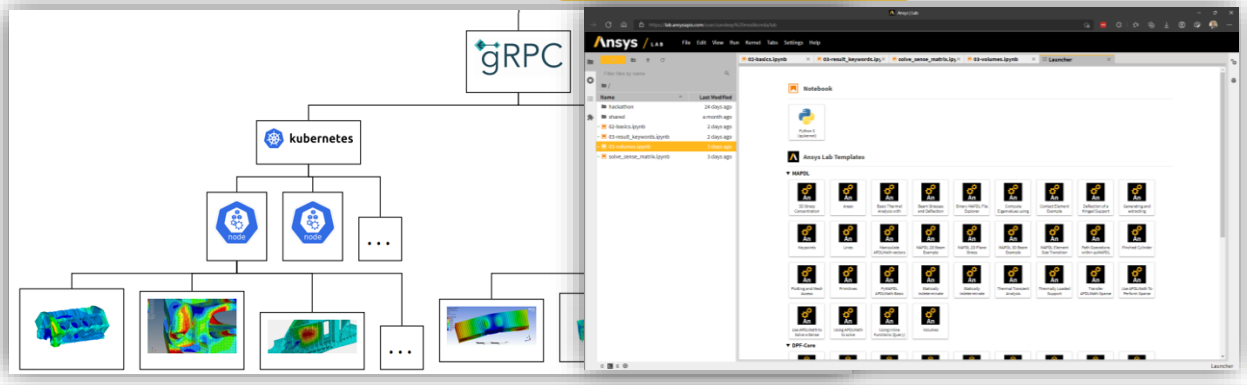




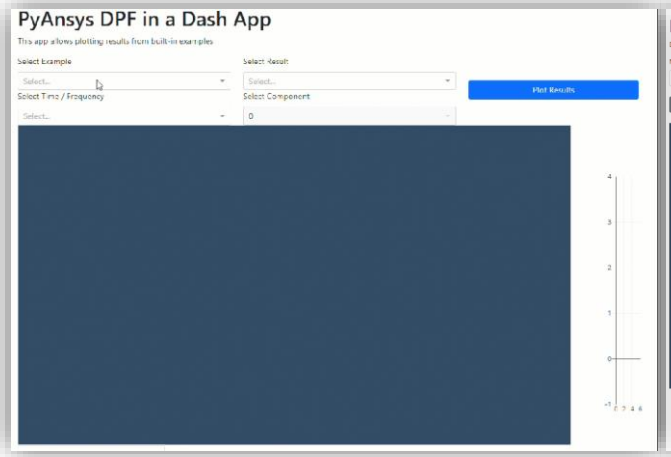
Multi-Physics workflow Automation

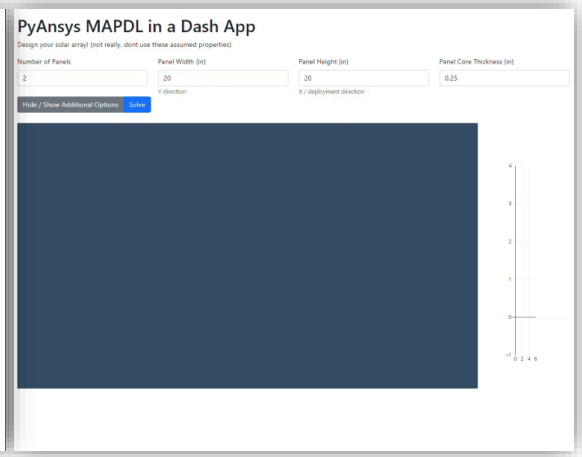
py[Ansys]

HPC Deployment



Web Applications





Ansys

Angle Bracket Optimization

Angle bracket has “dimple” to increase stiffness

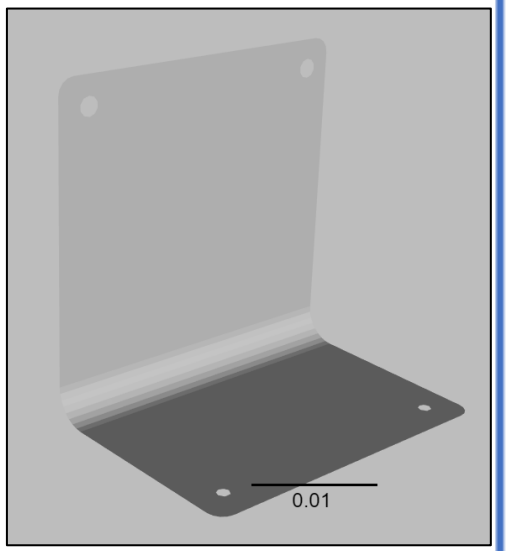
ML Example: Topographical Optimization using Reinforced Learning

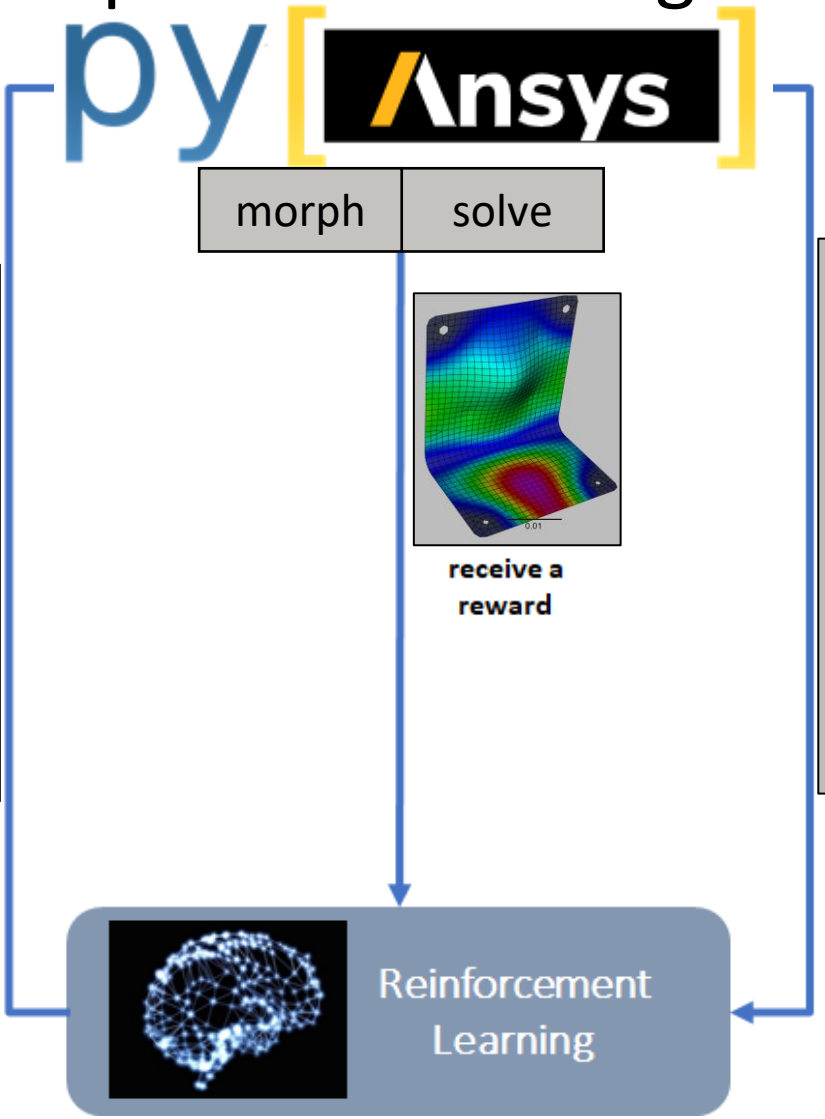
MDP Description

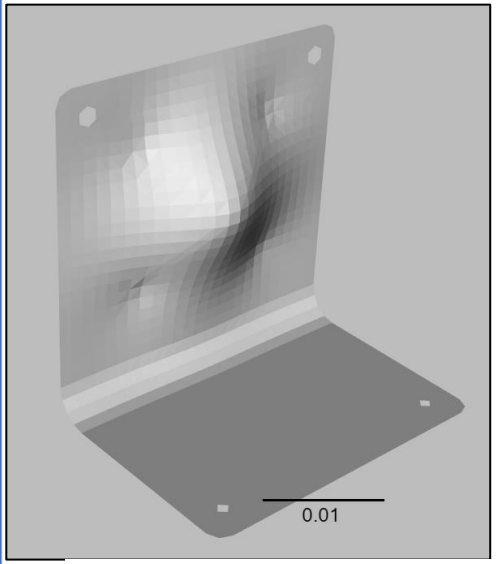
- Action:
  - morph point
  - morph direction
- State:
  - morph extent
- Reward:
  - Increase freq/stiff
  - penalty for distortion

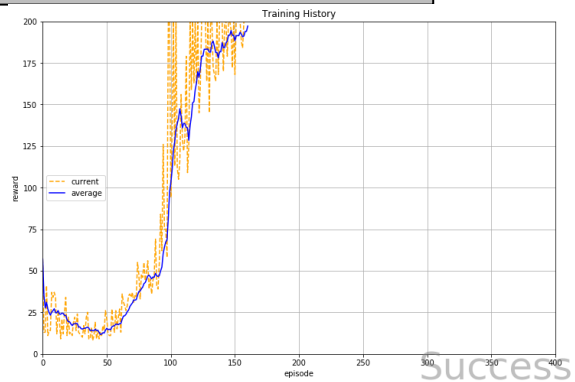
RL Algorithm

- Find sequences of morphing operations that maximize stiffness









Success!

Spray Dryer WebApp

Use Ansys Python Manager – Graphical Installation (windows)

Challenge/Need:

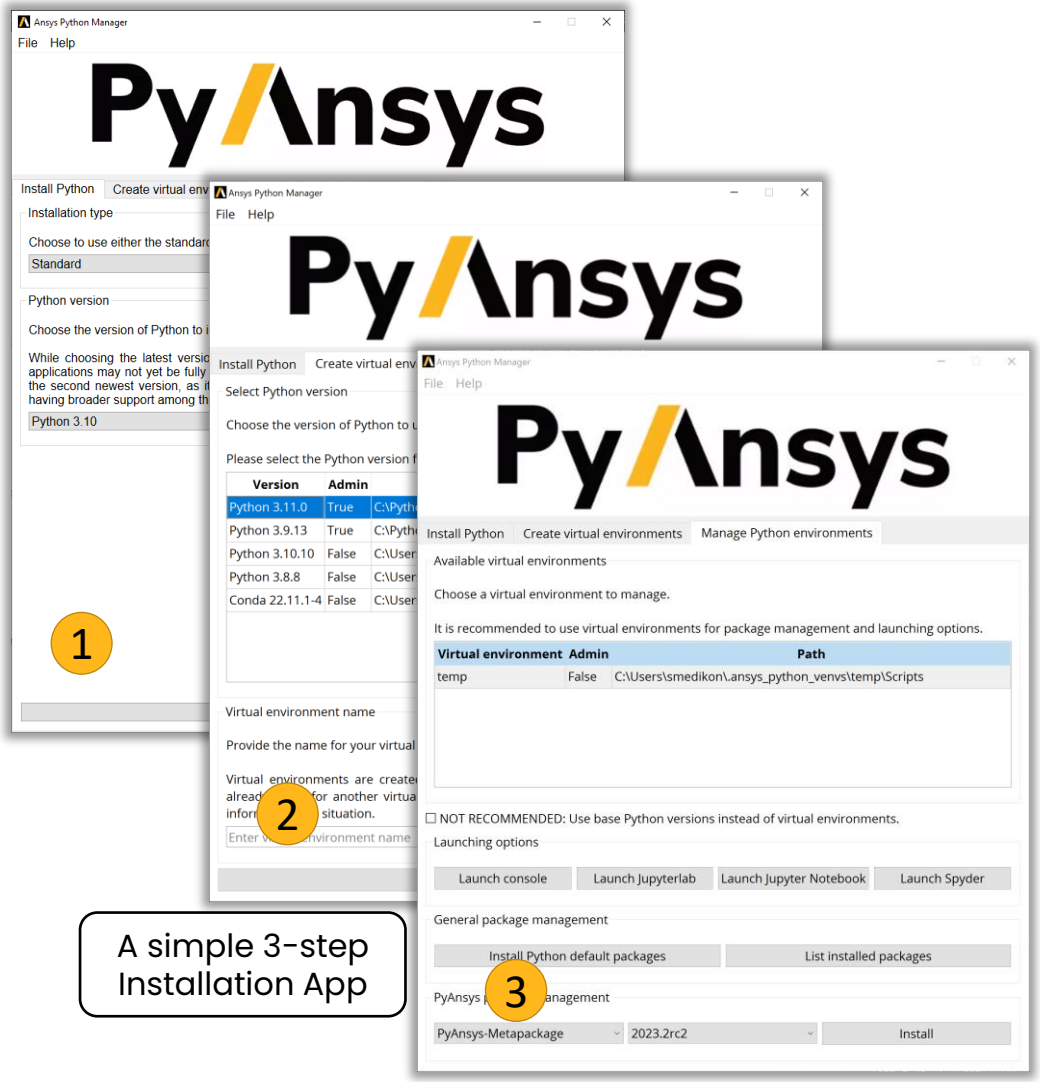
- **Anaconda GUI installer**: Transition to a **Commercial Business Model** was a major roadblock for Ansys Customers.
- Reduce Impact for Beginner & Intermediate Python users.

Solution/Tool Created:

- A simple 3-step installer app created in <1 month.

- Using Ansys Python Manager:
  Download it from:
  [Releases · ansys/python-installer-qt-gui (github.com)]

[https://developer.ansys.com/ansys-python-manager]



A simple 3-step Installation App

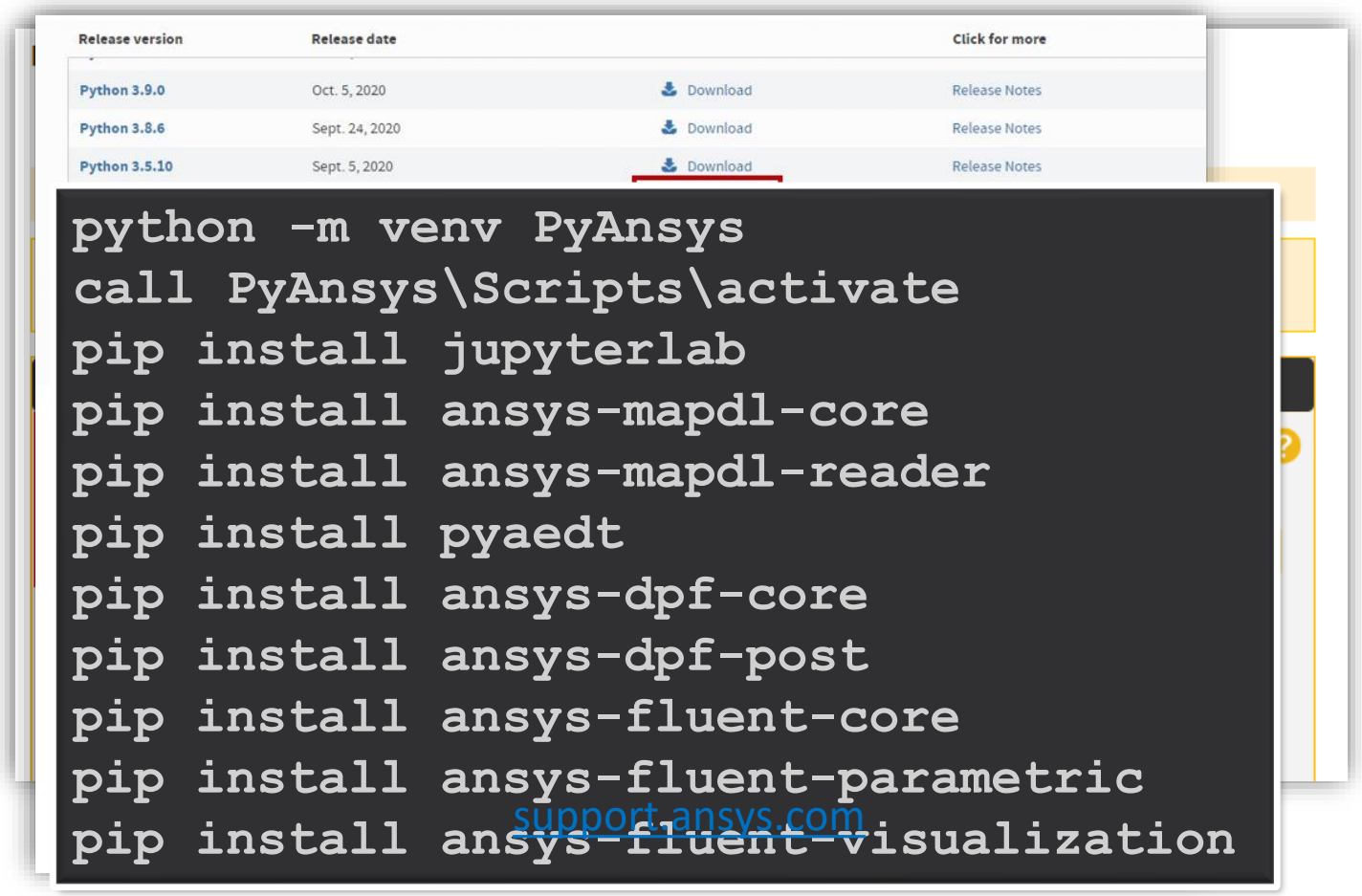
How to get started with PyAnsys (local installation)?

3 Step Process:

1. Install Ansys

2. Install Python and an IDE of your choice (Spyder, Jupyter, Jupyter Lab, …)

3. Install PyAnsys modules



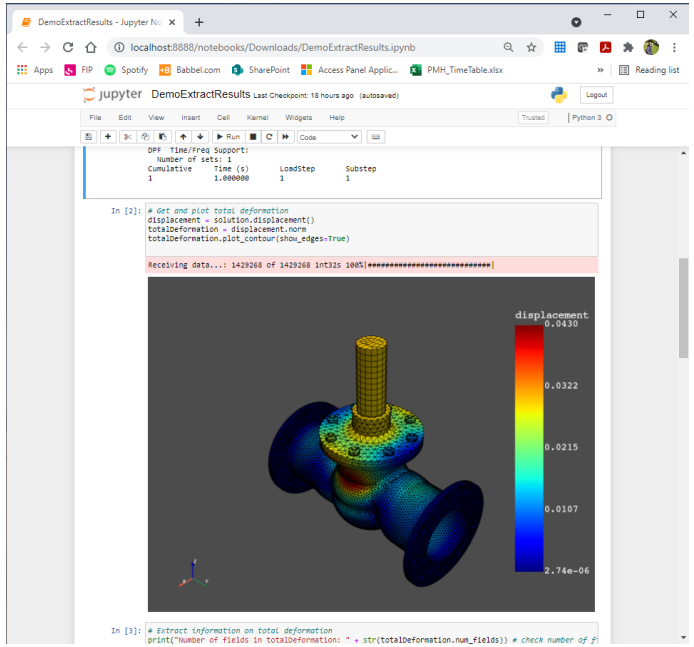
support.ansys.com

python.org/downloads/

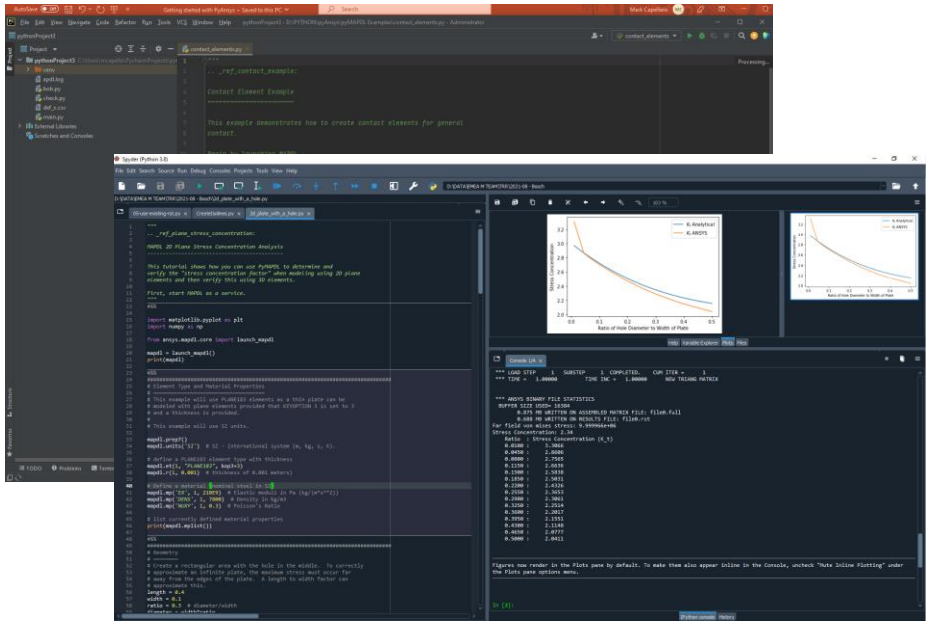
What's the UI like?

No specific UI. User's choice.

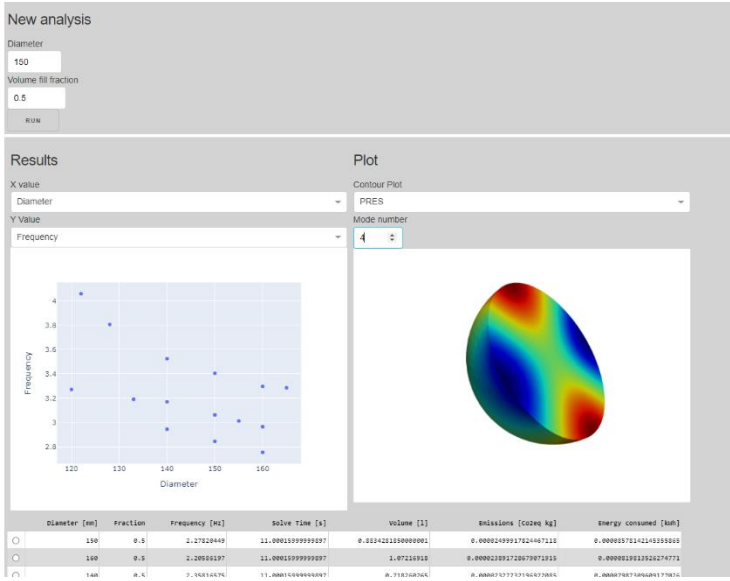
- PyAnsys is a set of tools to interact with Ansys products from a Python interface
- Any Python interface can be used (even the Windows Command Prompt !)
- Using an IDE (Integrated Development Environment) is recommended to write and maintain code.
  Many IDEs are available: Spyder, PyCharm, VS Code, Visual Studio, ….



Code can be written from a web-page thanks to Jupyterlab



An IDE can be used (PyCharm, Spyder, …)

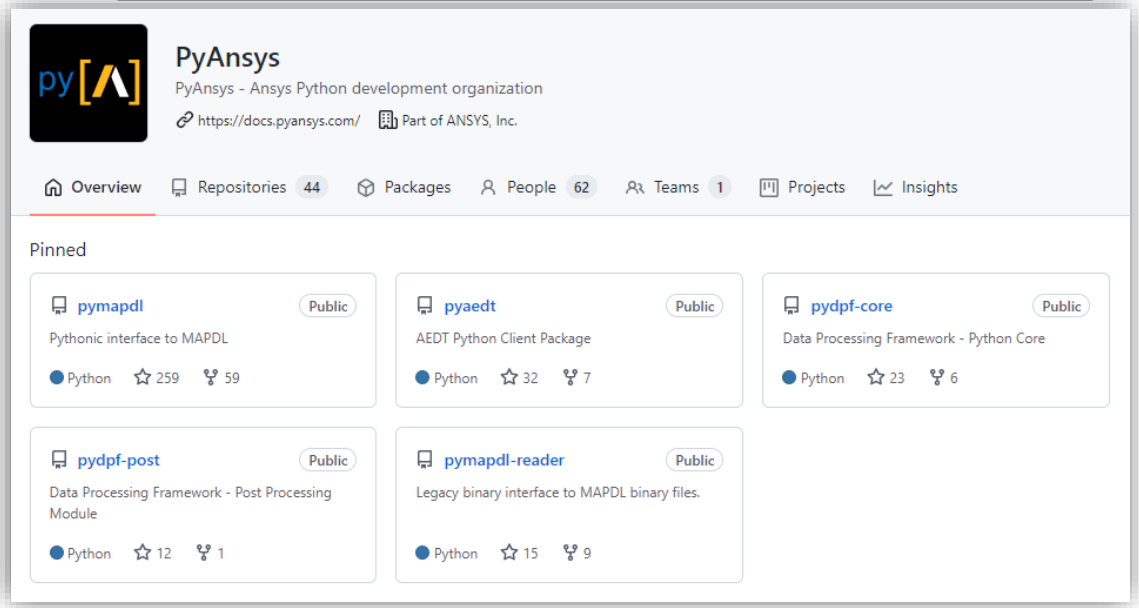


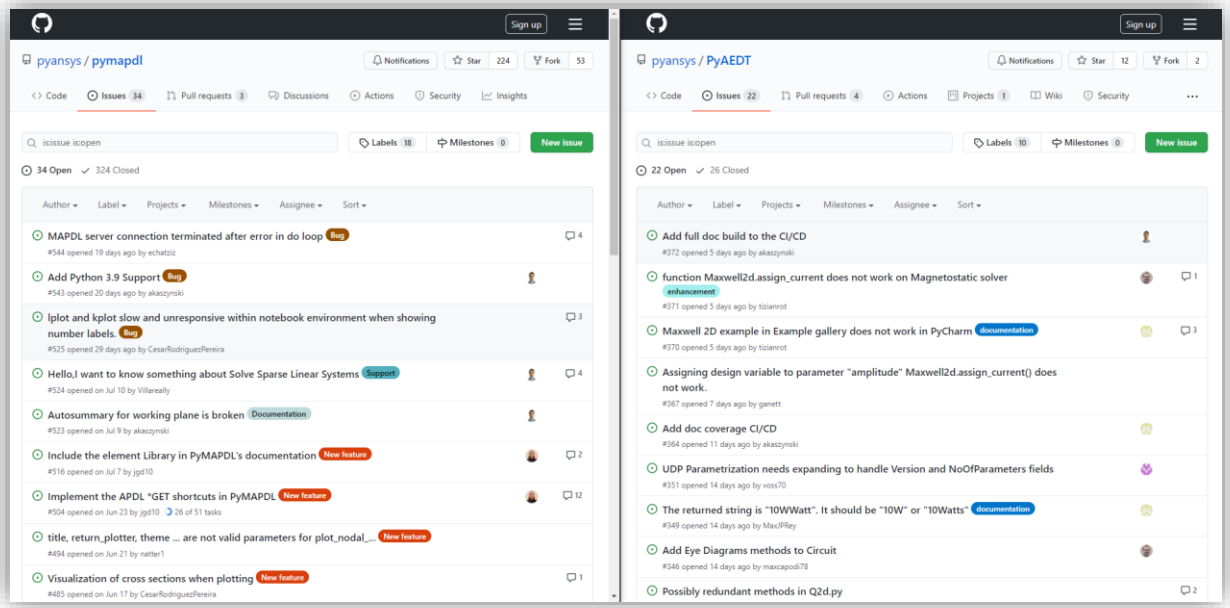
User can also develop apps by coding both the front end and the back end

Documentation & Help?

Source Code: https://github.com/ansys/pyansys

Contribute/Post-Issues directly on Github:

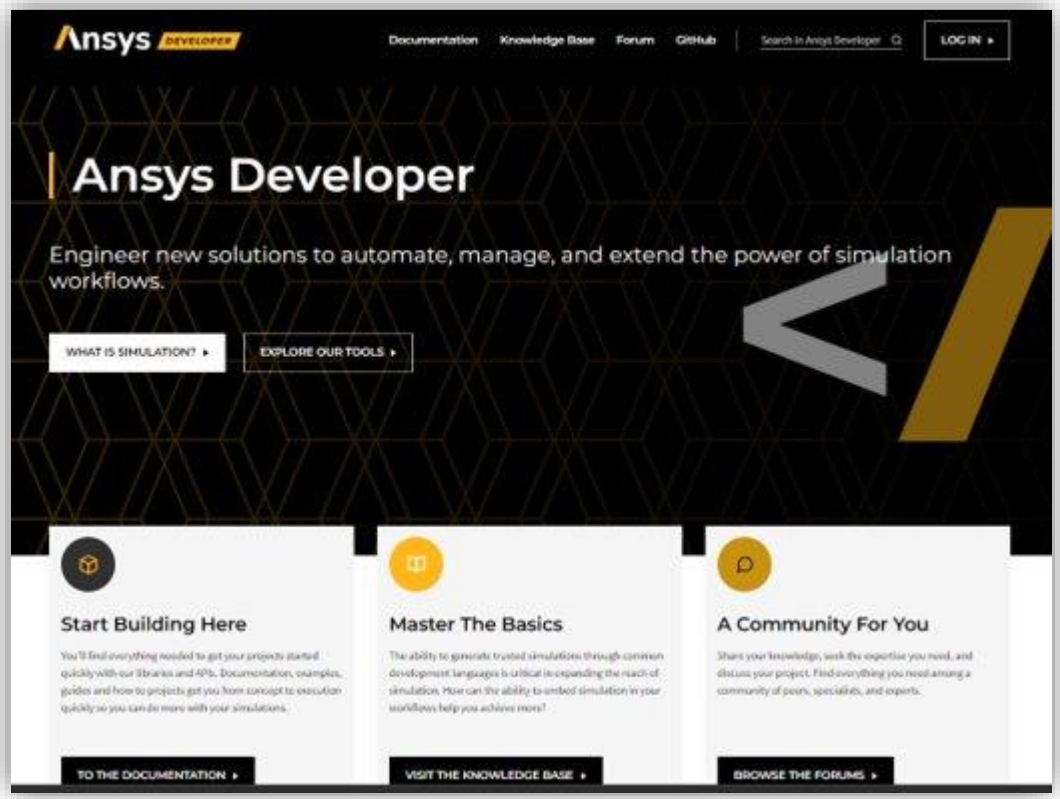


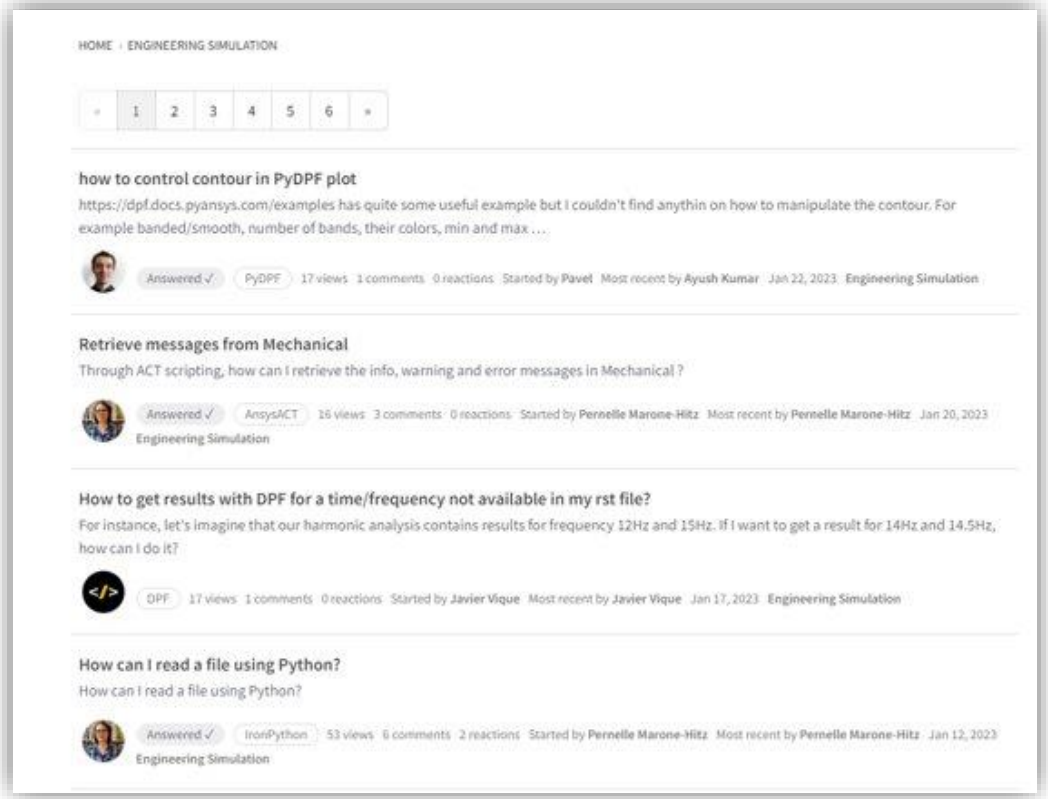


Documentation:

- Access from GitHub or use these Direct links:
  - https://docs.pyansys.com/
  - https://mapdl.docs.pyansys.com/
  - https://dpf.docs.pyansys.com/
  - https://aedt.docs.pyansys.com/

Need additional help? Use the Ansys Developer Portal and Developer Forum

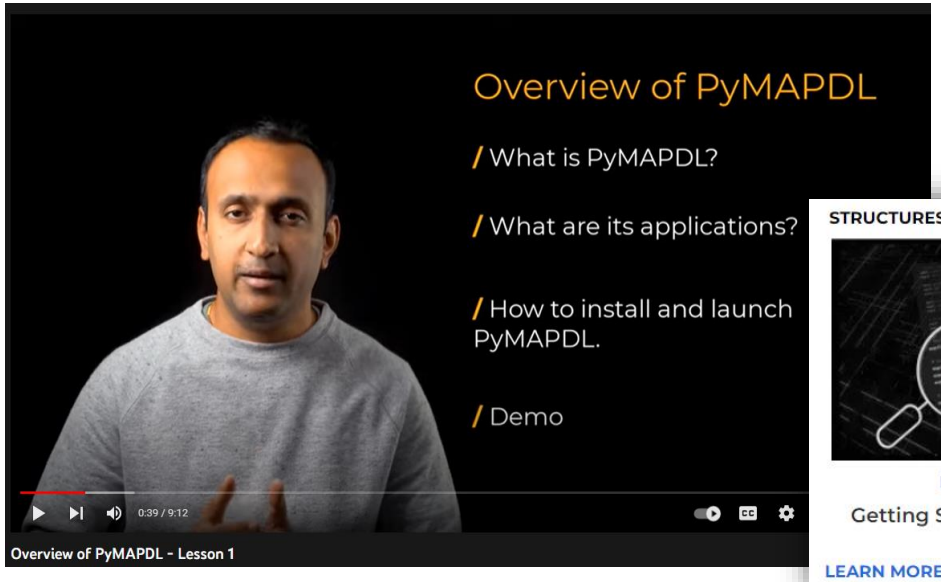




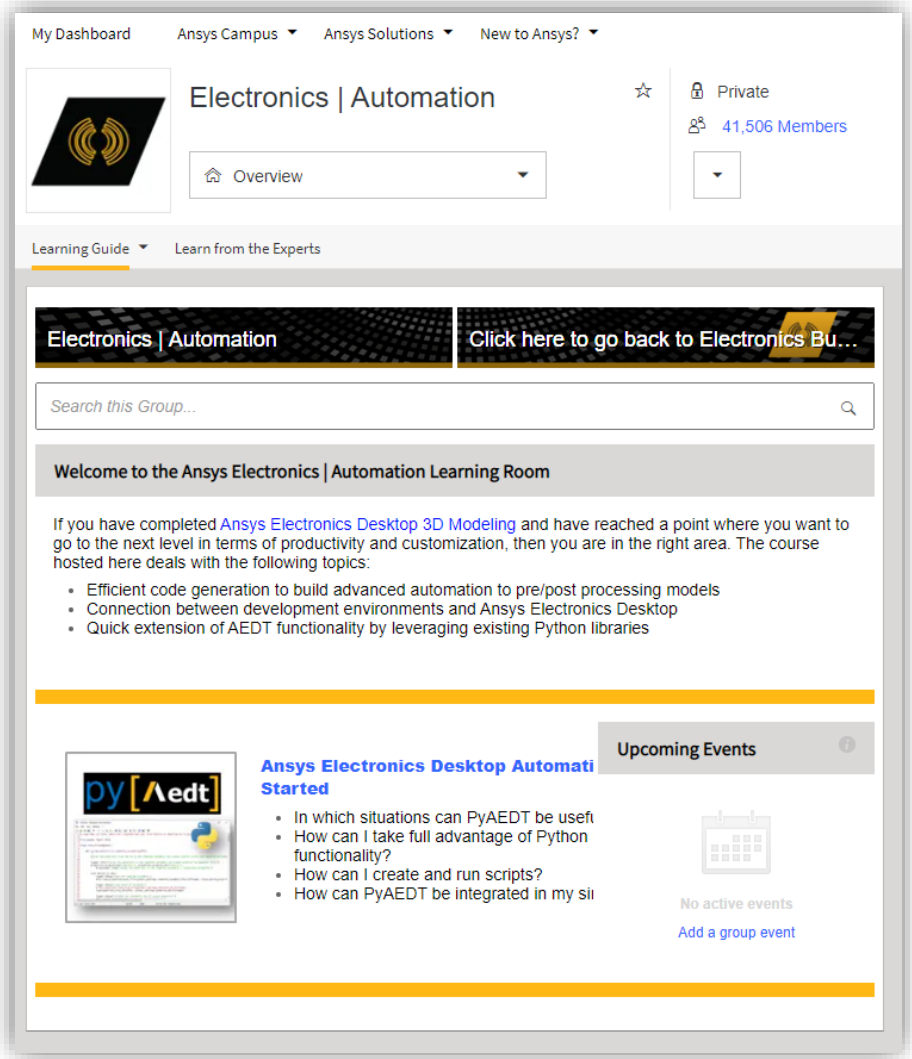
Ansys Developer Portal:
https://developer.ansys.com/

Ansys Developer Forum:
https://discuss.ansys.com/

Trainings (AIC & ALH)

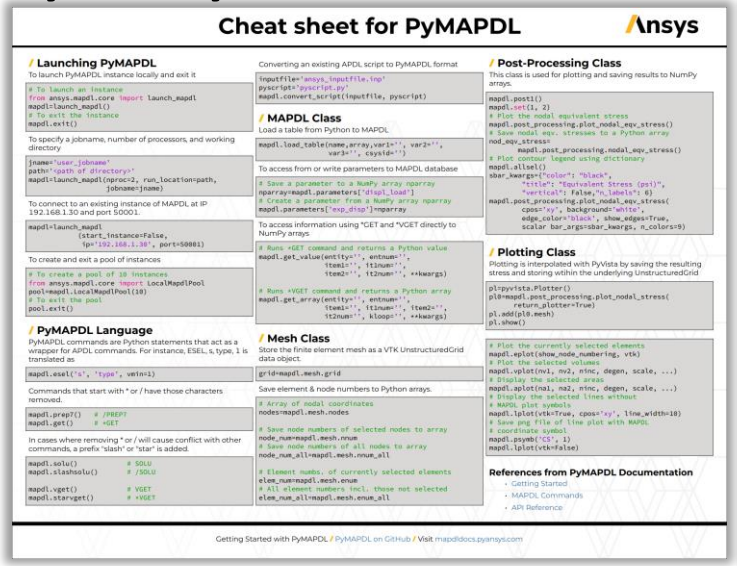


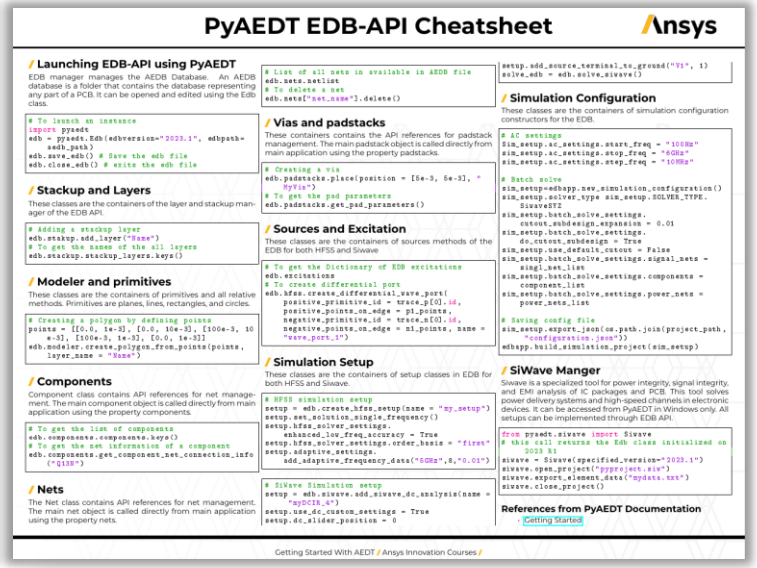
Ansys Innovation Courses (AIC)

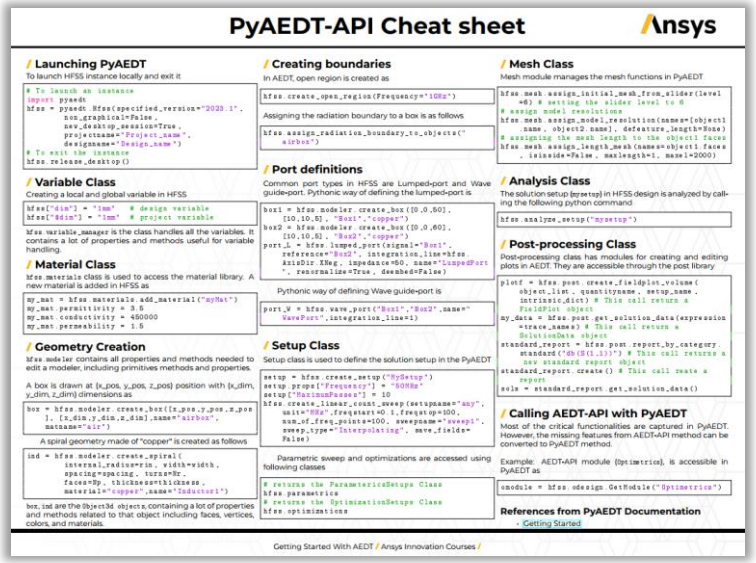


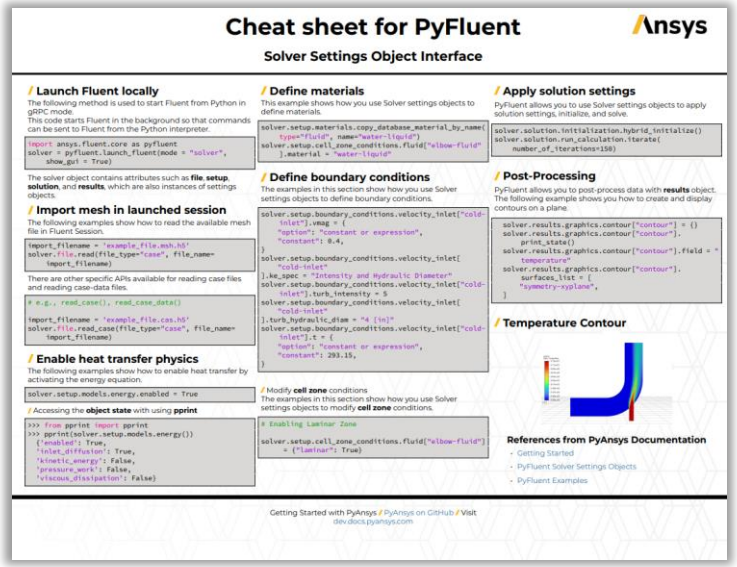
Ansys Learning Hub (ALH)

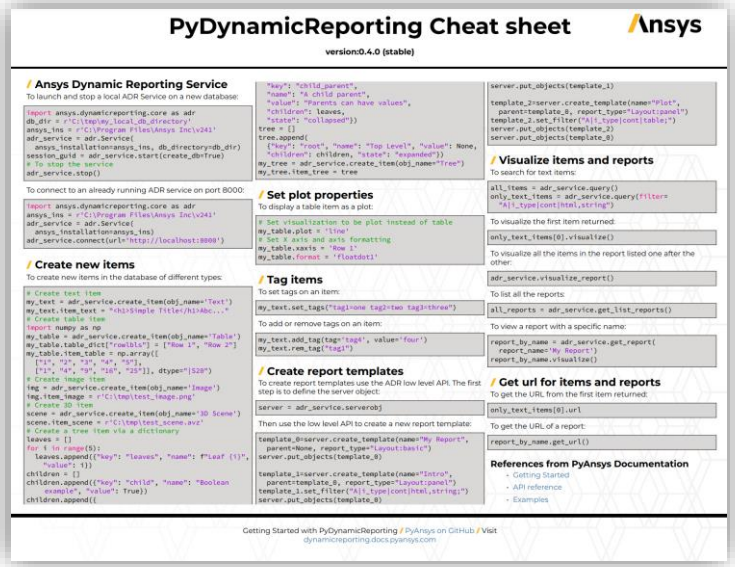
PyAnsys: Module Cheat Sheets











And many more on….
cheatsheets.docs.pyansys.com
or
developer.ansys.com

Testimonials – Research Community

Pyansys tremendously helped by providing an **easy** way to procedurally read and process data directly from Ansys without any human intervention. Combined with Python's enormous data analysis and plotting capability this helped me automate extraction of knowledge from 100s of giant Ansys simulations and saved a lot of expensive human hours

Ph.D. Student

You can **easily** retrieve node stresses and temperatures. Then outside, there is a material database which you can retrieve yield strength based on temperatures, and finally you can calculate factor of safety. And perhaps you can plot it. Think about you have many analyses in different locations and you can automatize this process

Aerospace Engineer

I was trying to avoid exporting my results to csv and then importing back into python. [I] started to look also into the "controller" features, which I came to like a lot.
I also contributed to pyansys once, to support reading transient analyses with a reduced number of DOFs […]. As far as I remember it was written in C, which I don't "speak".

Ph.D. Student

Quite **easy** to embed the simulation in different environments. I also have to say, how glad I am that the plotting functionality in pyansys is so much faster and intuitive than its APDL counterpart

Undergraduate Student

data extraction data transforms plotting

automation solver embedding user contributions

Conclusions

- PyAnsys introduces a **paradigm shift** in how Ansys simulation tools will be used going forward. Ansys is the '**first**' simulation software provider to introduce such **dynamic interaction** with its products.

- PyAnsys has been deployed on **GitHub** and is **controlled by ANSYS**.

- Ability to separate **Pre-processing**, **FE model** and **Post-processing** from outside the Ansys environment is a strength that PyAnsys has and will help us **deploy**, **maintain** and **scale** for applications across various industries.

- **AI/ML Community** can easily integrate ANSYS physics capabilities into their processes.


Ansys