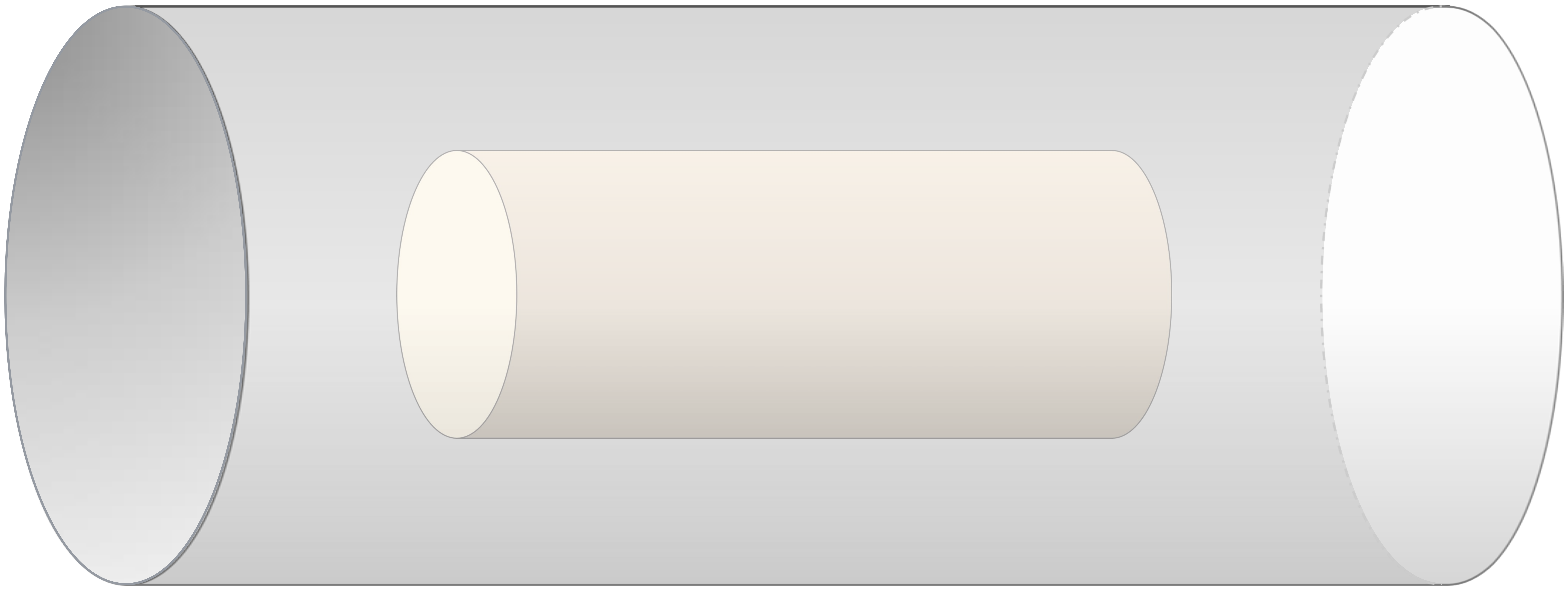
Big Data out of CFD

Piotr Prusiński, MSc, Eng.
Supervised by Prof. Sławomir Kubacki

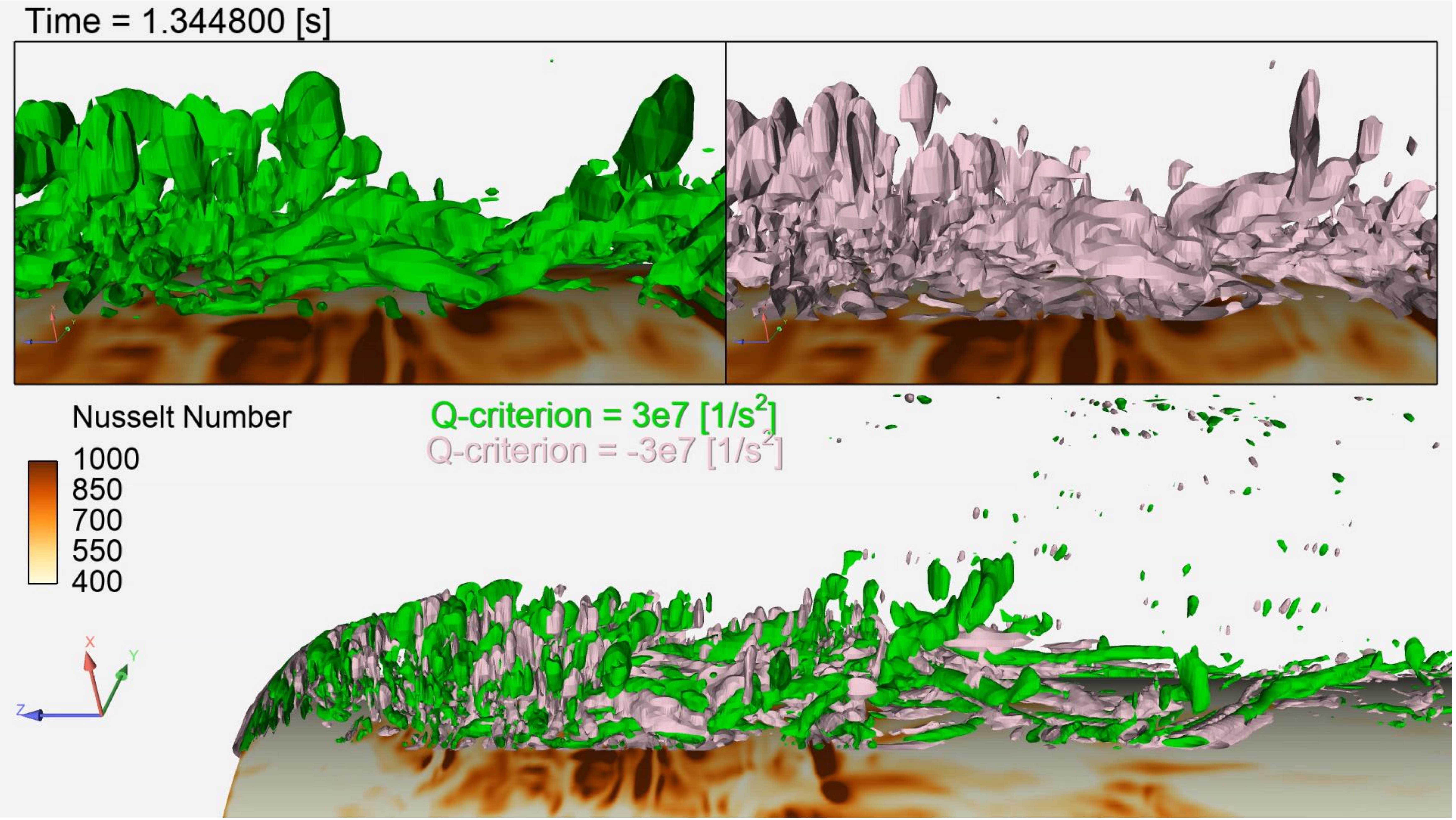
NATIONAL CENTRE FOR NUCLEAR RESEARCH ŚWIERK

Just to remind, what we are dealing with…



- Easy geometry = piece of cake, right?

Possible scenario



1. Developed turbulent flow, passing smooth pipe of inlet section, carries some weak streamwise voritcal structures. Those structures become condensed and hence stronger when approaching flat front of bluff body.
2. When passing a leading edge, they shift their shape into rings along the maximum strain rate surface, getting even stronger due to toroidal vortex pair.
3. The maximum strain rate surface breads Q-event structures.
4. Some fraction of vortical structures detach from the surface close to vena contracta (probably) due to ejection events.
5. At the same time, sweep events burst local heat transfer when hitting inner rod surface.
6. When detached, strong turbulent vortical structures start to reorient thier axes towards streamwise direction and elongates.
7. Elongated structures of alternate (+/-) rotataion pattern imprint either on fluid temperature isosurfaces and on inner rod wall surface.

Did you know…

- that the size of a setup file to run such a simulation is just:

11.5 GB

- but the size of a results file/checkpoint (one timestep) is:

93.3 GB

Now imagine…

- Since the current timestep is:

135 440

- I could expect to have in total:

11.5 + 135 440 × 93.3 = 12 636 563,5 GB

= 12 PB (raw data)

What makes it so Big?

Domain Size × Number of timesteps

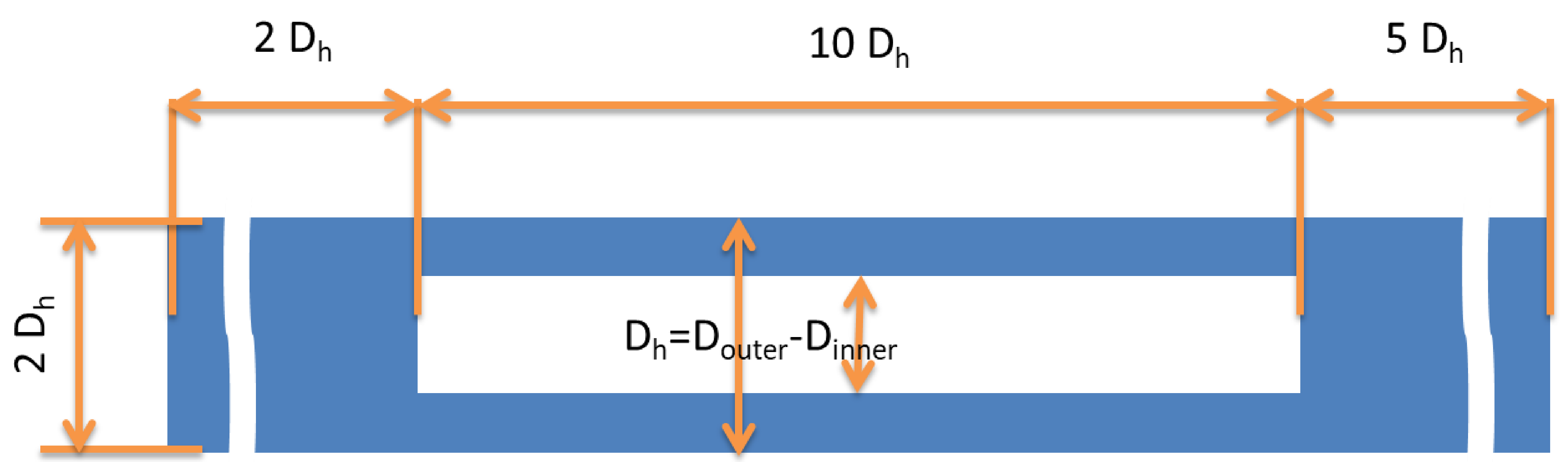
11.5 + 135 440 × 93.3 = ?

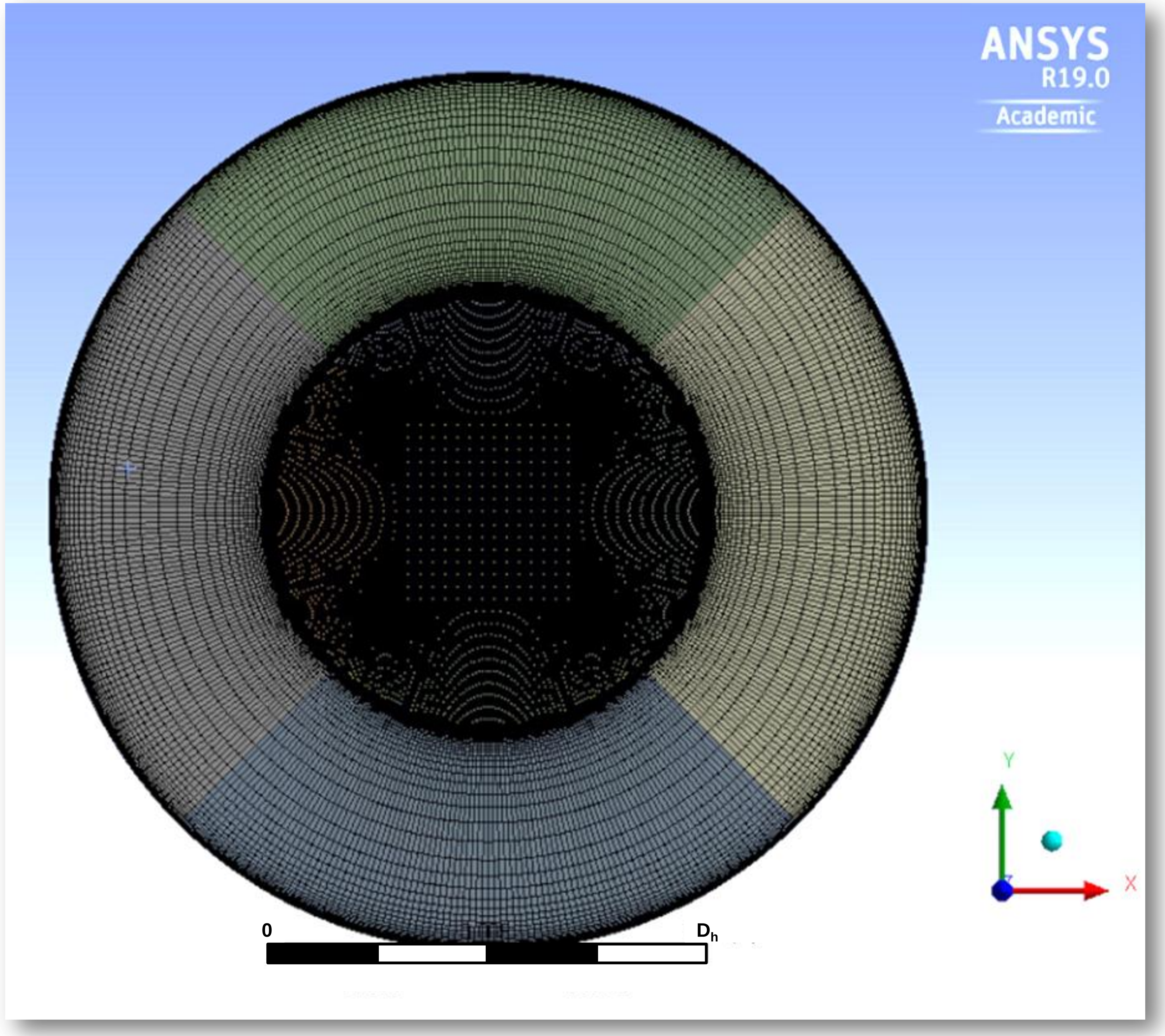
The question is:

- Do I really need so much data?
  - How big is the simulation domain and do I need such a space resolution?
  - How long the simulation should be? (simulation period)
  - Does every timestep counts or is valuable? (time resolution)

How big is the simulation domain?

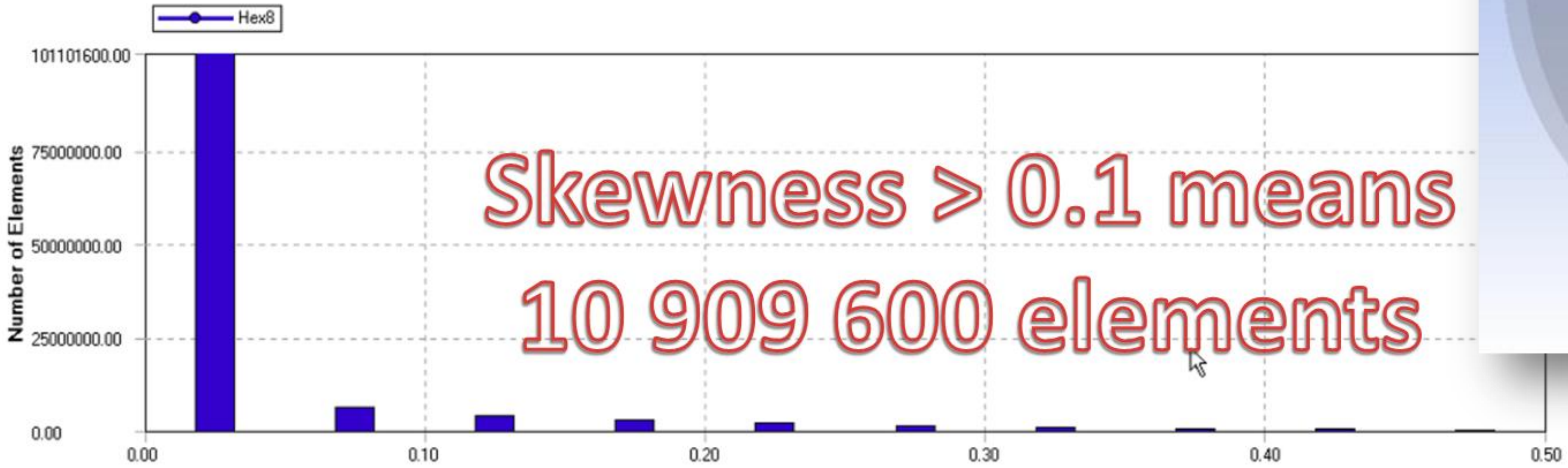
- Full 3D
  - 117 960 000 elements (hex)
  - 119 124 502 nodes

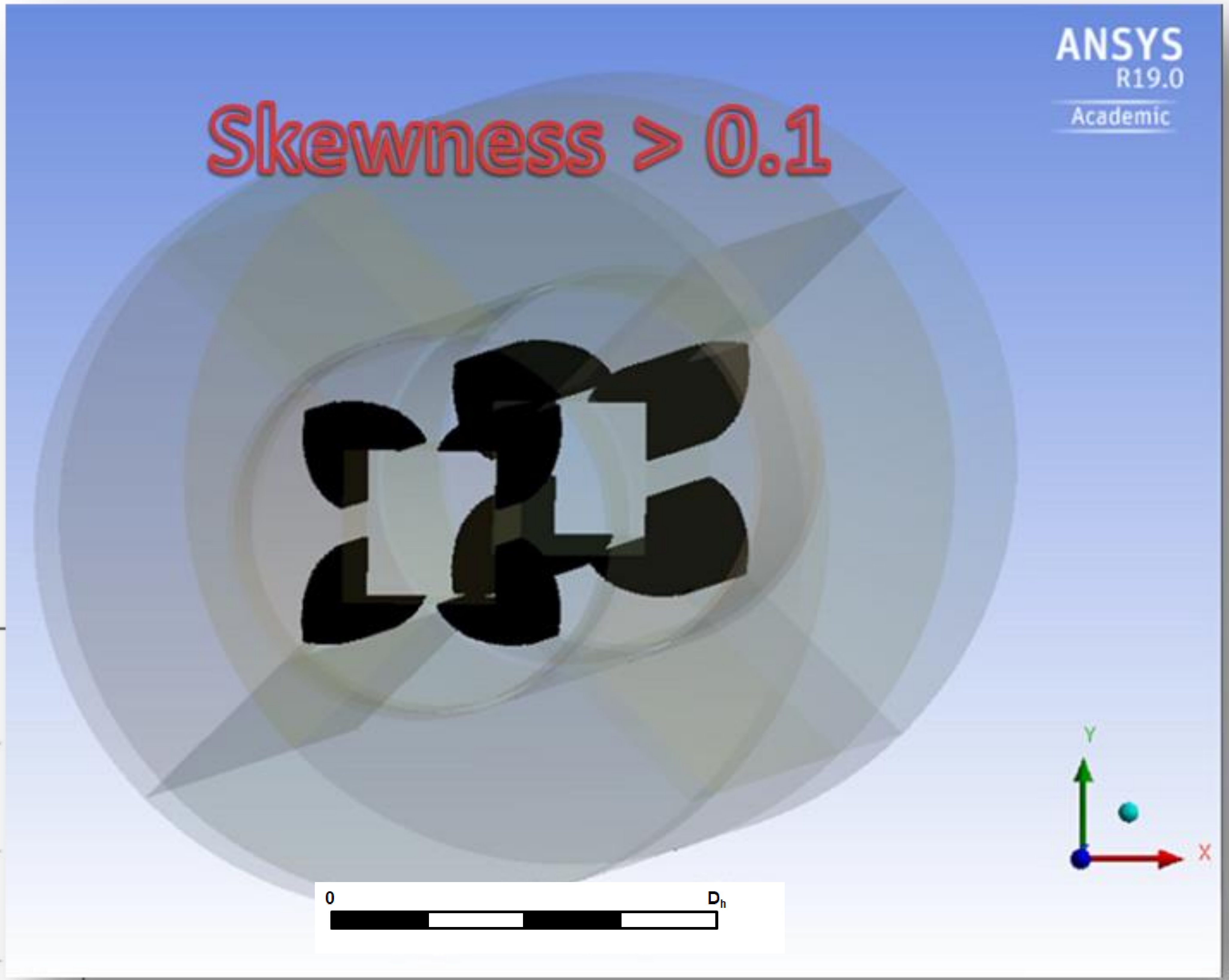




Skewness

- Skewness
  - Min 1.3e-10
  - Max 0.5
  - Ave 2.9e-2
  - Std Dev 6.5e-2

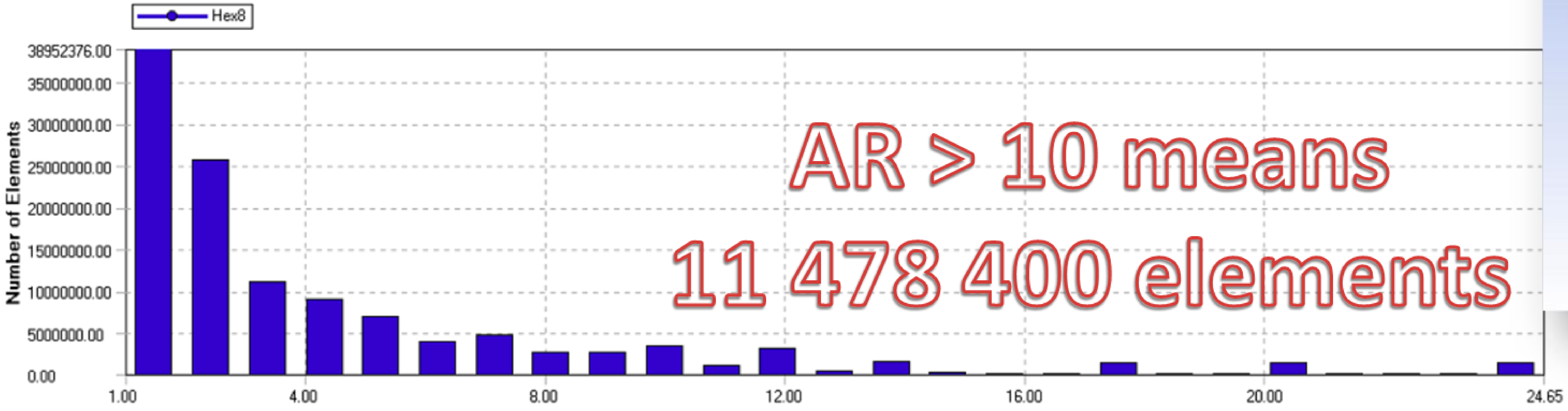


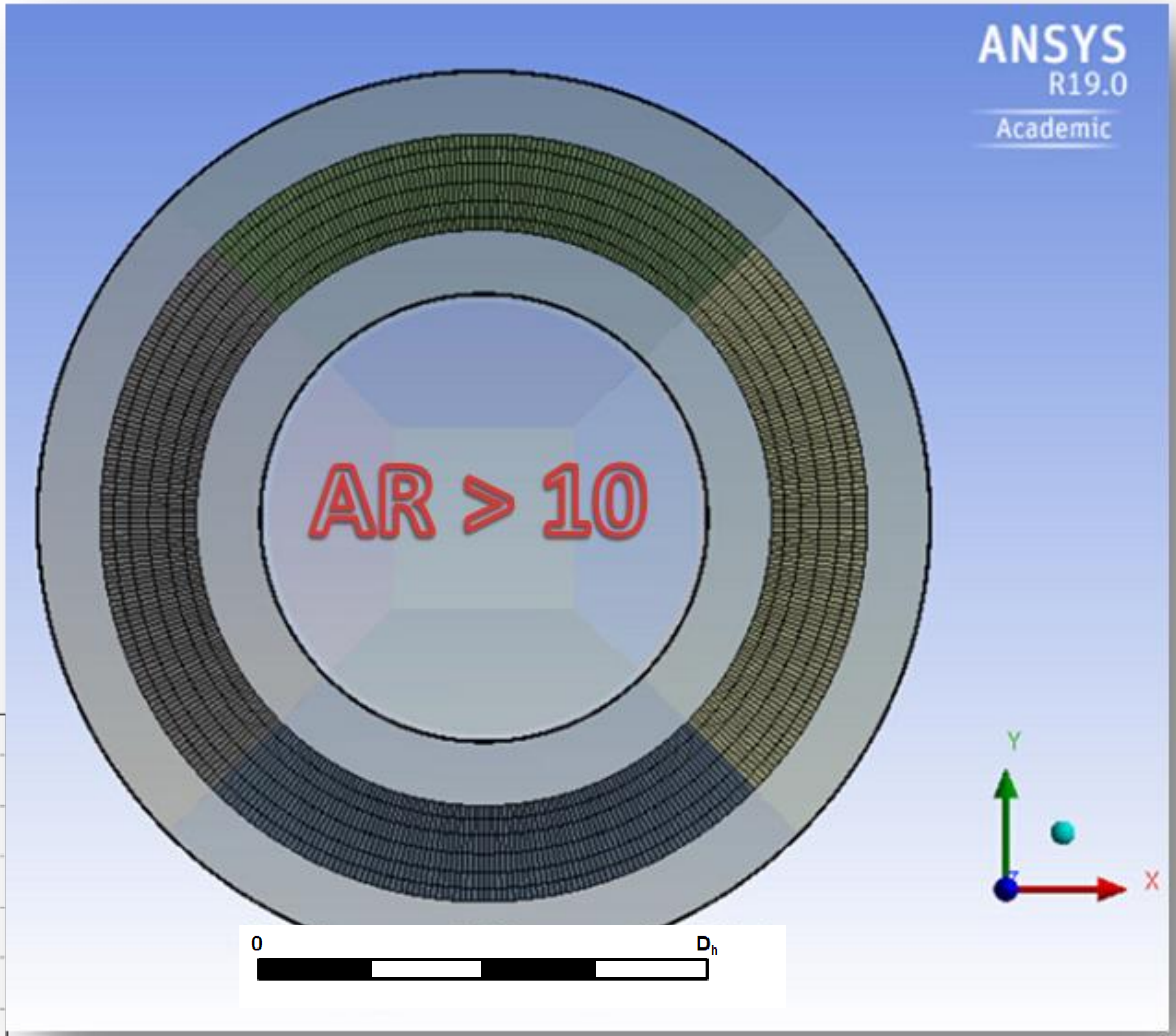


Excellent	Very good	Good	Acceptable	Bad	Unacceptable
0-0.25	0.25-0.50	0.50-0.80	0.80-0.94	0.95-0.97	0.98-1.00

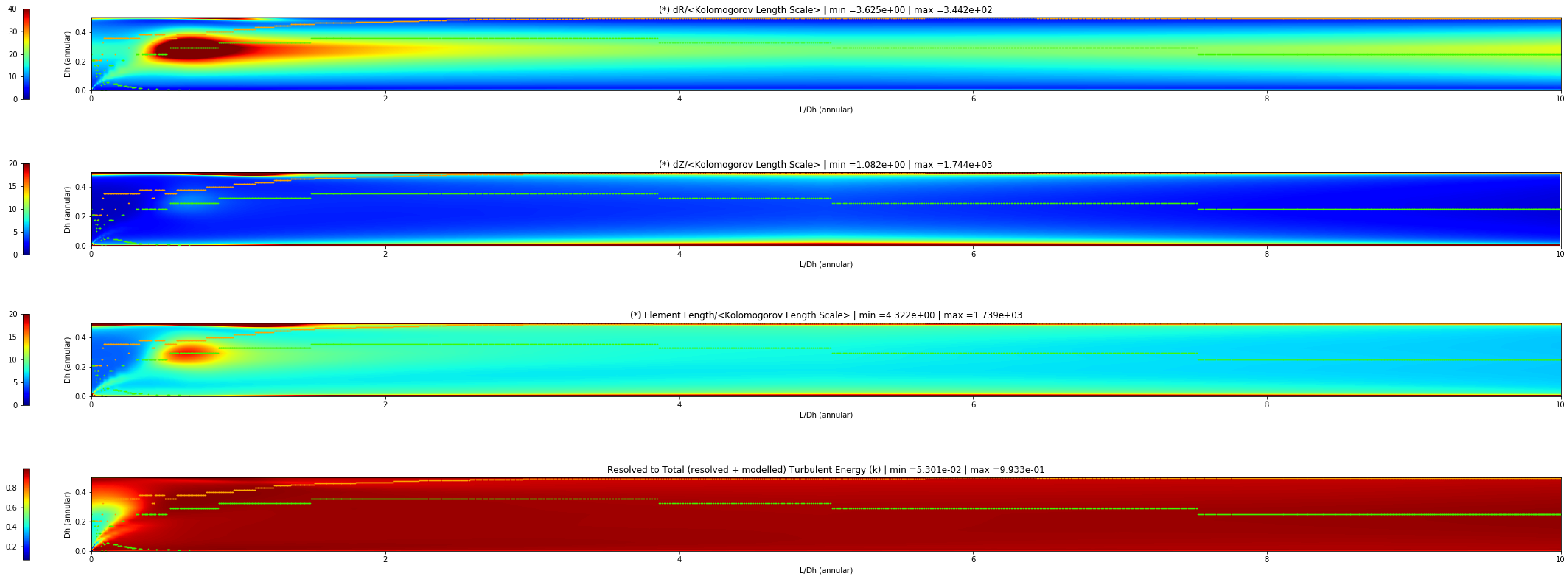
Aspect Ratio

- Aspect Ratio
  - Min 1.0001
  - Max 24.653
  - Ave 4.3921
  - Std Dev 4.3174





Do I need such a space resolution?



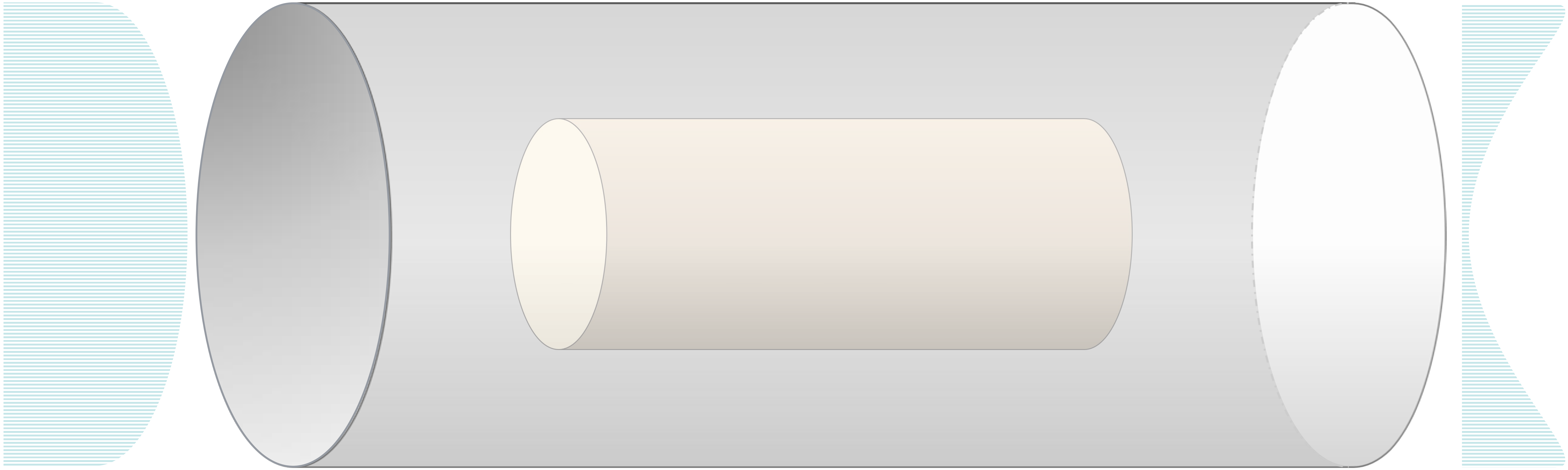
1st conclusion

- Based on Aspect Ratio and Kolmogorov length scale compared to Radial element size, some refinement still could be applied. Mainly by adding extra layers, but…
  <u>1 more layer in radial direction means 1.3M new extra elements!</u>
- Based on Resolved to Total Kinetic Energy Ratio in the annular part of domain, presented discretization seems to be sufficient to conclude the observations.

- ***The current mesh resolution is already a (reasonable) compromise!***

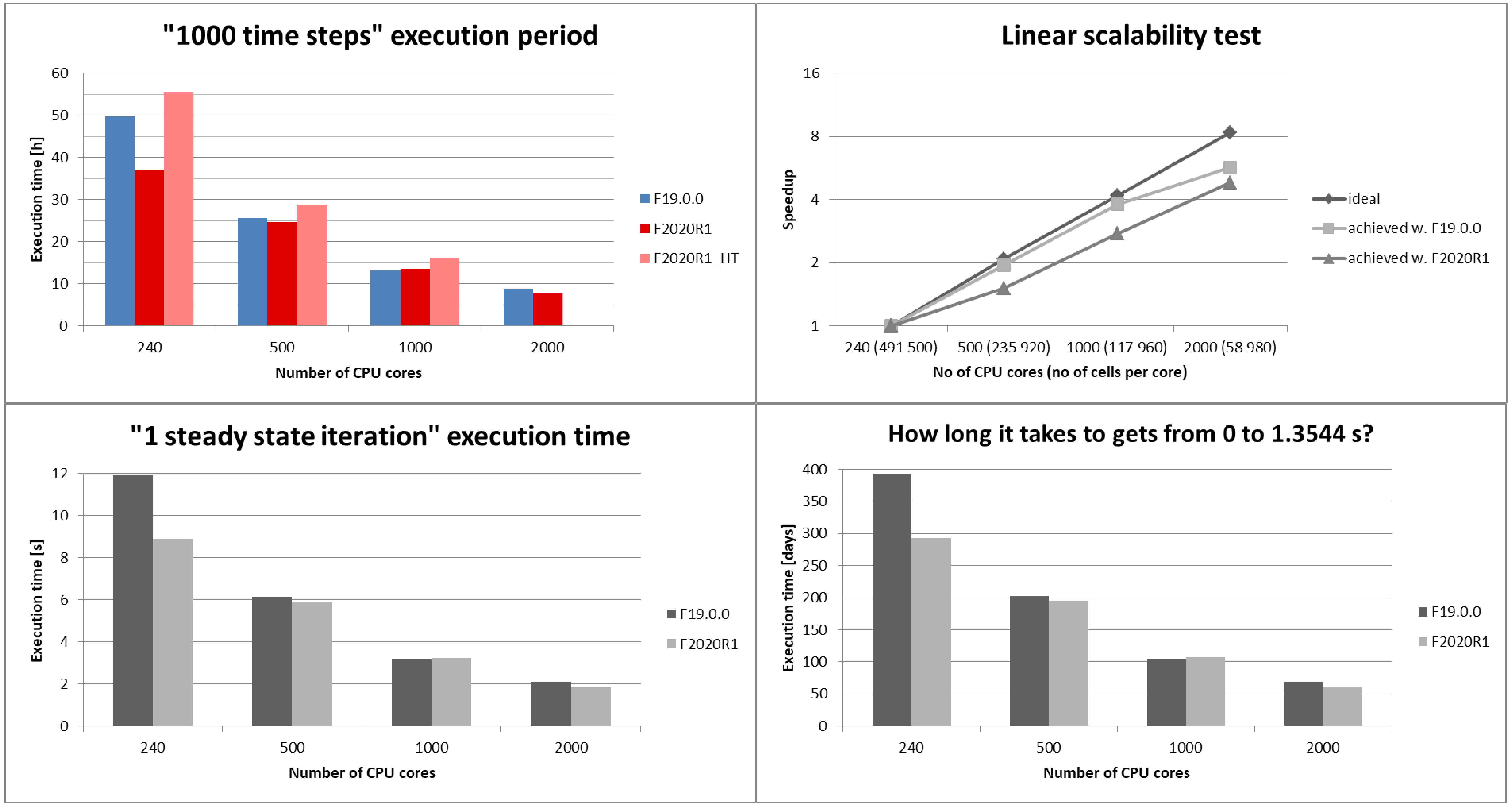
How long period of time do I really need?

- 135 440 timesteps ≈ 4.55 flow volumes replaced (FVR)



BTW. How long it really takes to get where we are?

- Up to now LES went through 135 440 timesteps (2 849 100 it)
- Computational time is tangled with number of software licences available, i.e. 1 licence = 1 CPU core

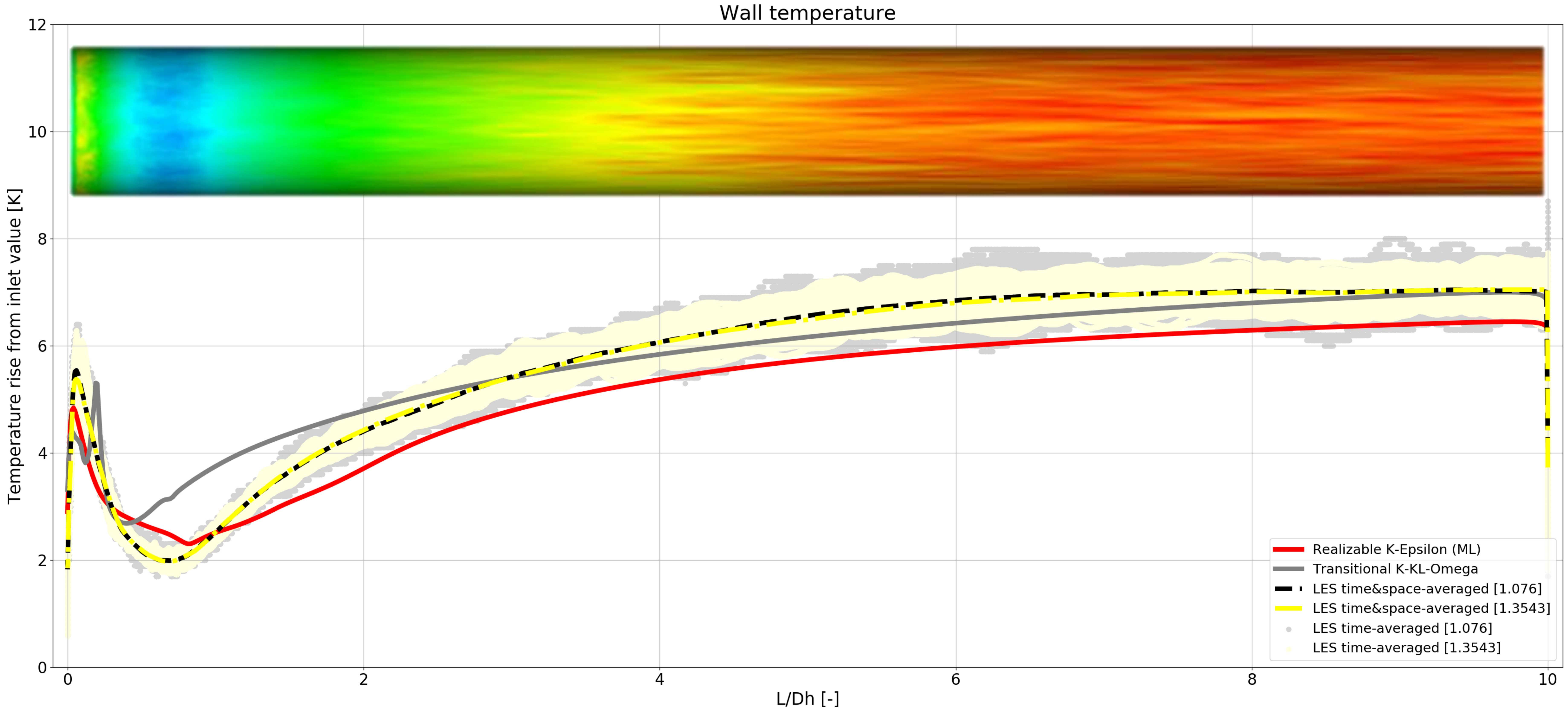


What about averaging period?

- Time-averaging done in two independent periods:

1)
   - **Starting from 3.29 FVR**
   - **For the next 0.33 FVR**

2)
   - **Starting from 4.07 FVR**
   - **For the next 0.49 FVR**



2^{nd} conclusion

- Before running the time-averaging, we need to have at least 1 FVR past. More is always advised.
- As we could see, even 0.5 FVR was not enough to get time-averaged results without further post-processing. It is estimated that about 2-3 FVR would be necessary to make it possible, but we would not recommend this anyway for economical reasons.

- **It is well depicted that home-made Python script for space averaging for two different and independent periods of simulation bring the same answer. In other words, we could have shorter simulation, but we could not prove it validity without prolongation.**

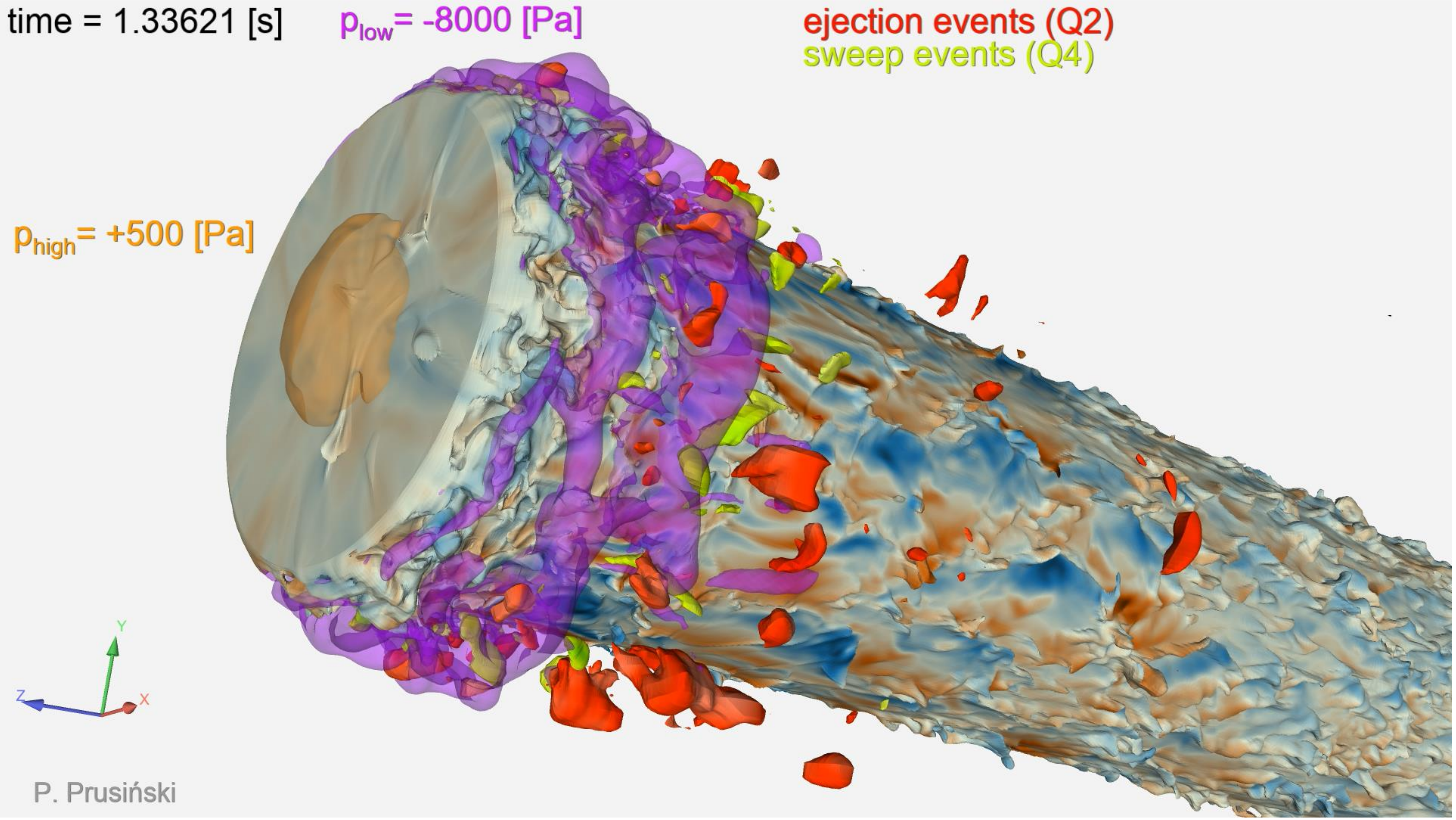
Do we really need to keep every timestep?

- Depending on what we are looking for:

YES	NO
If we want to spot some instant flow structre or its pattern or even behaviour. In most cases this is done by means of visualisation. When making an animated visualisation one needs certain time resolution to get smooth motion of such a structure.	If we are going to base only on the time-averaged results. It is done after every timestep, so in fact we need only the final results file.

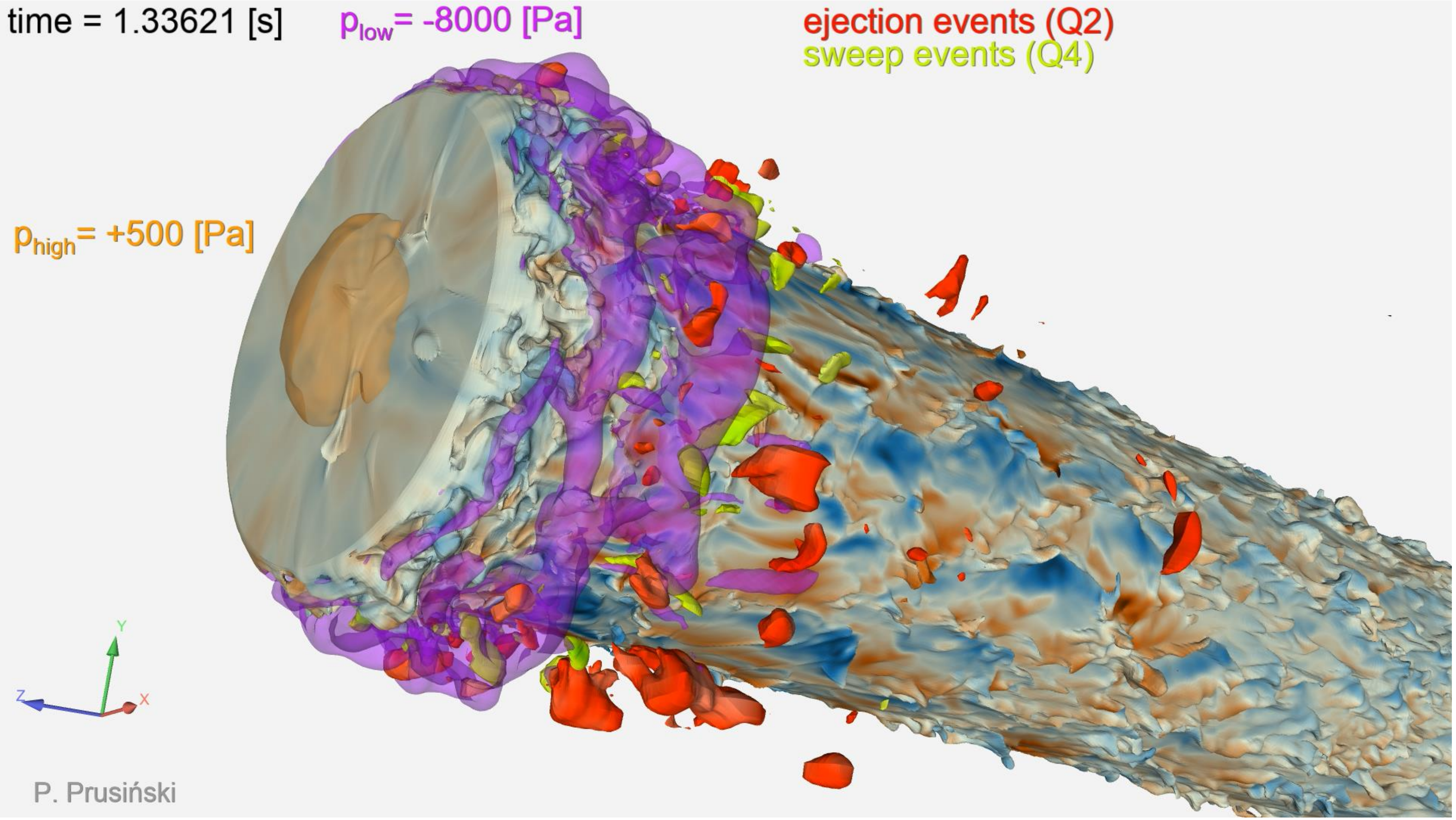
NARODOWE CENTRUM BADAŃ JĄDROWYCH ŚWIERK

So what is the frequency necessary in this case?



- Dependce on phenomena observed!
- Δt = 0,005 FVR HiTR and NooV (10) means 2.1 TB!

How many variables were used in this visualisation?



13.7 GB CGNS

Extracted directly:
- Time
- Coordinates
- Scalars:
  - Instant Pressure
  - Time-Averaged Pressure
  - Temperature
  - Time-Averaged Temperature
- Vectors:
  - Instant Velocity (Cylindrical)
  - Time-Averaged Velocity (Cylindrical)

Computed in post-processing:
- Scalars
  - Lamianar-To-Turbulent Threshold
  - Q2 event
  - Q4 event
- Vectors
  - Velocity Fluctuations (Cylindrical)

3rd conclusion

- The front of oscillating pressure is much quicker than flow structures and this is a proof that the pressure oscillation has minor (or no) impact on heat transfer.
- Presented movie is really smooth. In this case it seems we could reduce the number of frames to every 2nd one. And if we get rid of pressure consideration, we can even reduce it to every 10th one. Although one should remember:
  - No one knows upfront all the details
  - During visualisation one can always judge that specific combination of data needs refinement in time. This is easily accessible when data where collected, but if not - one needs to run another simulation with requested time resolution.

- **Visual and animated context of collected data cannot be underestimated, so we cannot relay just on averaged data, when explaning physics.**

The question is:

- How much data did I produce to understand the physics?

252/203 TB

since end of 07.2018

- This is about **50 times less** comparing to original estimation. How did I make it?

Native data are quite „heavy”

- So I used them only as a rerun checkpoints. How often?

Once a day

- What was the checkpoint frequency?

Every 500th timestep

11.5 GB + 135 000 / 500 × 93.3 GB = 24.6 TB

But wait?!

11.5 GB + 135 000 / 500 × 93.3 GB = 24.6 TB

- That is even 10 time less than 252 TB that I achieved, right?
  - Yes, indeed, the disk space occupancy is mainly due to post-processing files (and some side branches of the experiment)

Post-processing setup issues (1/3)

- Let’s make a simple test:
- We have a following files setup:
  - LES_r05_19.cas = 11.5 GB (HDD)
  - LES_r05_1e-9_1035000.dat = 93.3 GB (HDD)
- RAM (when .cas read into Fluent): 174 GB
- RAM (when .cas + .dat read into Fluent): 355 GB!

- 355 GB of RAM required to post-process the data!

Post-processing setup issues (2/3)

- Even the fastest workstation at NCBJ could not handle our BigData due to ANSYS Fluent's poor RAM management and also due to obvious RAM limitations!
- We had to find a way to not get stucked until we get more RAM.

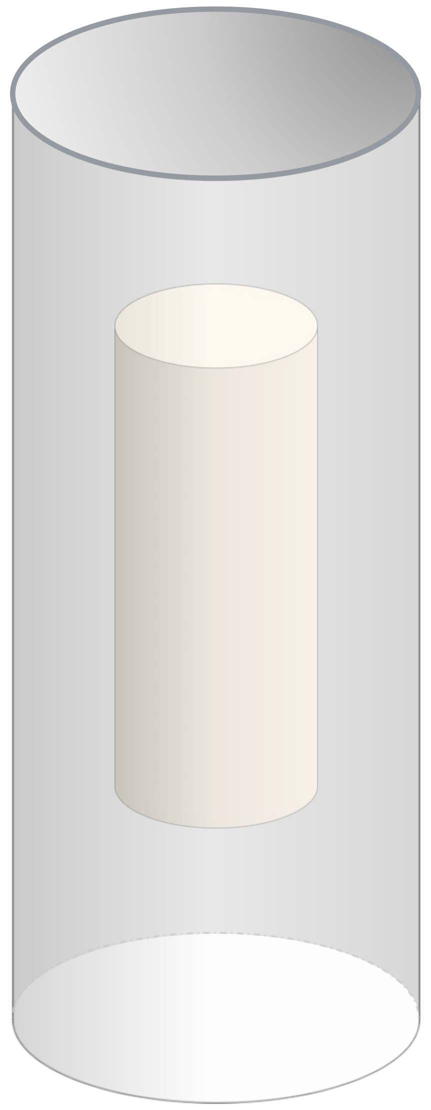
Post-processing unit name: GPU107

Server info:	
• Model:	ASUS - ESC4000 G3 Series
CPU info:	
• Model CPU:	Intel(R) Xeon(R) CPU E5-2680 v3
• CPU / server:	2
• No of cores / CPU:	12
• No of threads / CPU core:	2
• Total No of cores / server:	24
• Total No of threads / server:	48
• CPU frequency (nominal/TurboBoost):	2500MHz / 3300MHz
RAM info:	
• Available RAM:	256/512 GB (normal/evaluation period*)
• Type:	DDR4
GPU info:	
• No of GPU:	2
• GPU type:	NVidia Tesla K80 (24GB VRAM)
• GPU unit:	GK210GL
• GPU driver version:	418.39
• CUDA driver version:	10.1
• CUDA compute capability:	3.5
Intel OpenCL:	
• Intel OpenCL library version:	16.2
• OpenCL compute capability:	1.2
Interconnections:	
• Interconnect:	1 x Ethernet (1Gbit/sec per port) 1 x Infiniband FDR (56Gbit/sec per port)

Post-processing setup issues (3/3)

- Fluent 19.0.0 input size (105M hex, LES): 10.2 (*.cas) + 87.6 (*.dat) = 97.8 GB
  - Saving a result (*.dat) file takes about 7 min (@ 900 CPUs)
- Fluent 2020/R1 input size: 4.5 (*.cas.h5) + 68 (*.dat.h5) = 72.5 GB
  - Saving a result (*.dat.h5) file takes about 28 min

88 field variables!



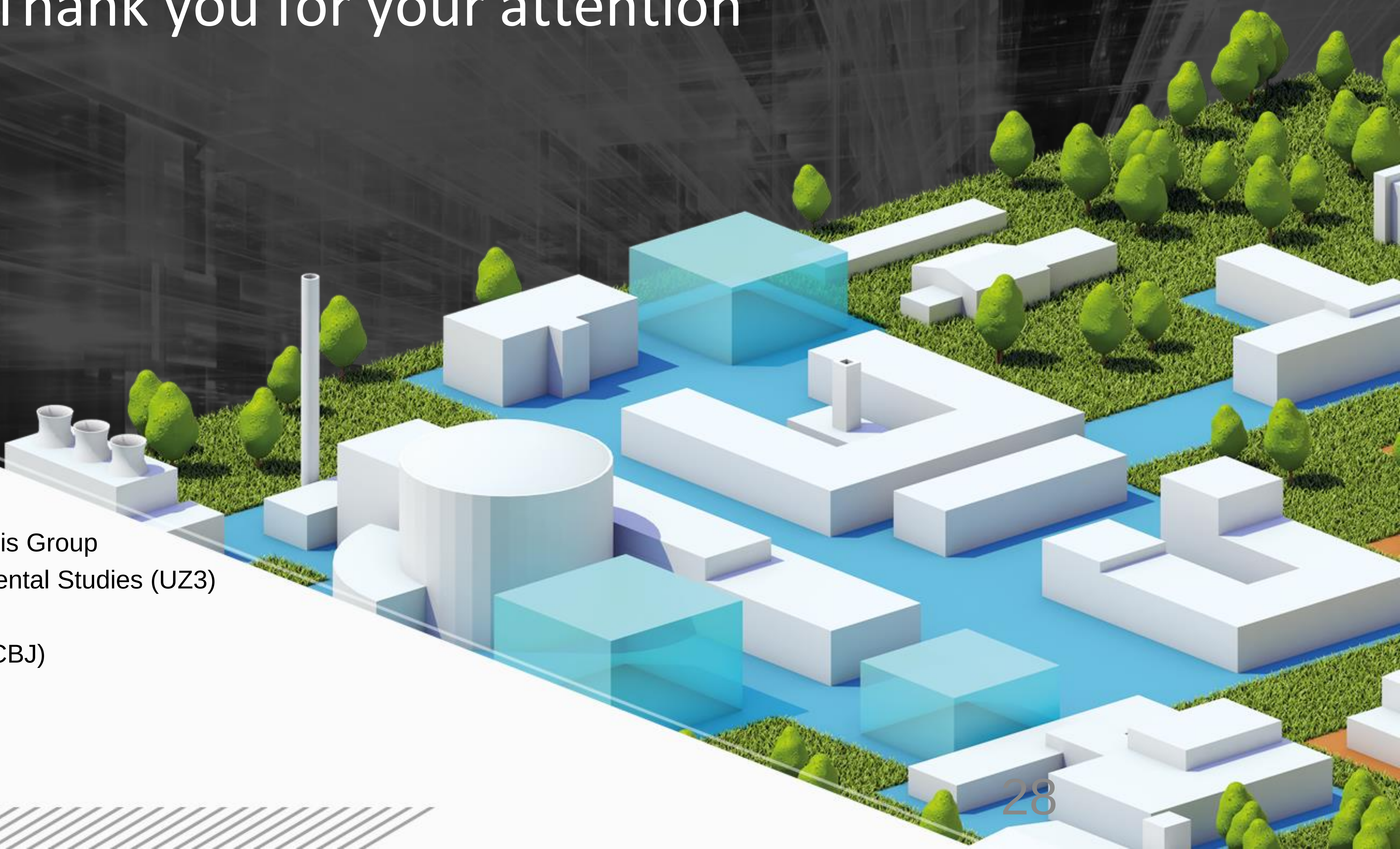
2020_R1 Fluent (time)				
physical CPU cores	cdat	cgns	plt	Averaged out of X samples
900	00:25:17	01:32:07	28:18:36	8
(pure mesh) 900	-	00:05:51	00:19:38	9
(no surfaces) 900	00:27:47	-	01:34:49	12
(no interior) 900	00:23:47	-	00:09:02	10
(interior only) 900	00:23:45	-	01:21:42	5
(inplc) 900	00:26:32	-	02:02:47	6
(inplc) 2000	00:18:18	-	03:59:13	5
460	00:10:59	01:15:20	22:21:09	6
220	00:13:44	01:11:32	19:16:59	7

2020_R1 Fluent (file size in GB)			
physical CPU cores	cdat	cgns	plt
900	142.74	79.60	44.45
(pure mesh) 900	-	9.57	6.05
(no surfaces) 900	142.74	-	39.36
(no interior) 900	56.63	-	0.81
(interior only) 900	141.0	-	43.6
(inplc) 900	141.0	-	43.9
(inplc) 2000	144.0	-	45.0
460	141.45	79.60	43.80
220	140.31	79.60	43.23

Final conclusions

- LES/DNS type of simulation always yields far more data than its physical counterpart in a lab experiment. Both can provide deep insight through BigData analysis.
- When trying to catch yet unexplained physics one should carefully plan every move, i.e. from size of a domain, through timestep resolution up to a choice of post-processing format.

Thank you for your attention



NATIONAL CENTRE FOR NUCLEAR RESEARCH ŚWIERK

Piotr Prusiński, MSc. Eng. | CFD Analysis Group
Division of Nuclear Energy and Environmental Studies (UZ3)
Departament of Complex Systems (DUZ)
National Centre for Nuclear Research (NCBJ)
A. Sołtana 7, 05-400 Otwock-Świerk
e-mail: piotr.prusinski@ncbj.gov.pl
tel: +48 22 273 11 26