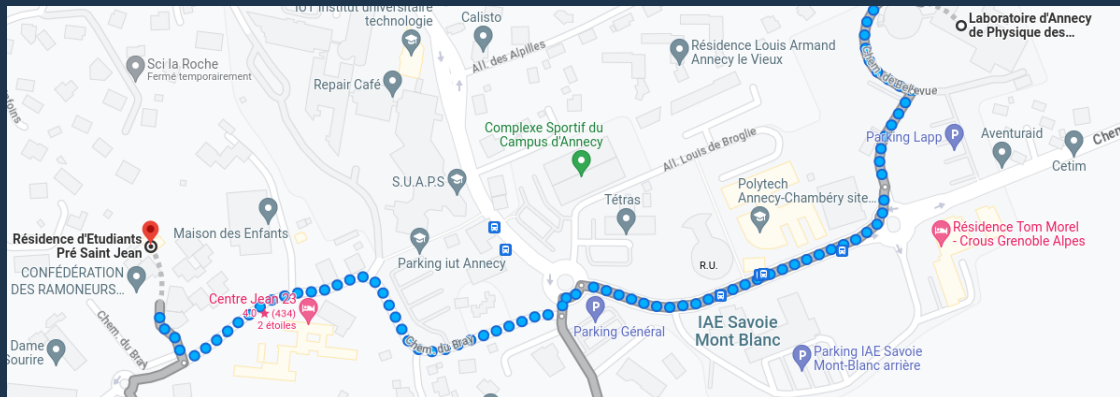


Plantage du décor

Pierre Aubert



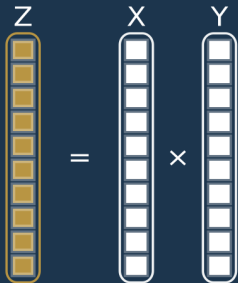
Introduction



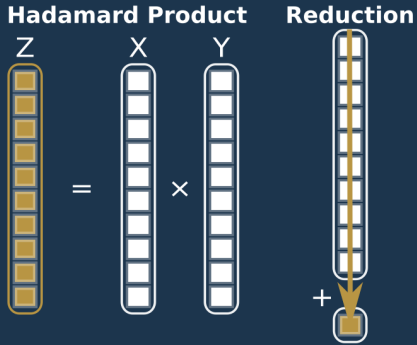
Simple Examples

Simple Examples

Hadamard Product

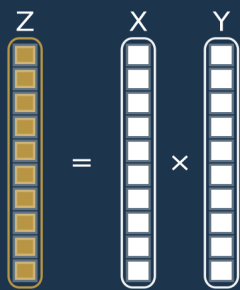


Simple Examples



Simple Examples

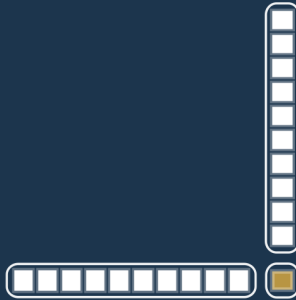
Hadamard Product



Reduction

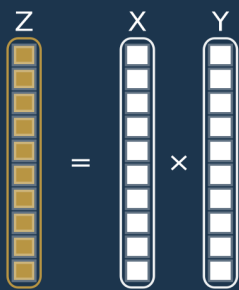


Dot Product



Simple Examples

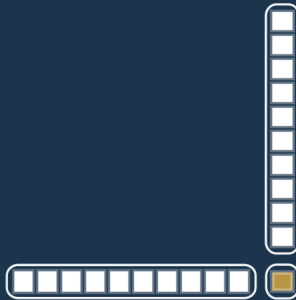
Hadamard Product



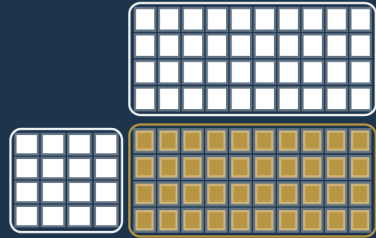
Reduction



Dot Product

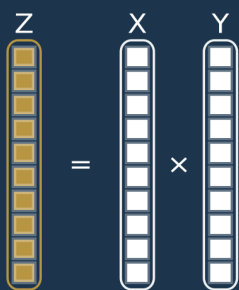


Matrix Product



Simple Examples

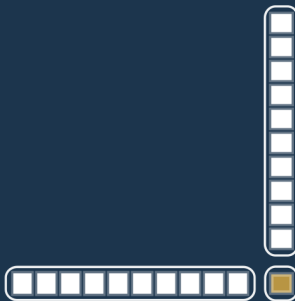
Hadamard Product



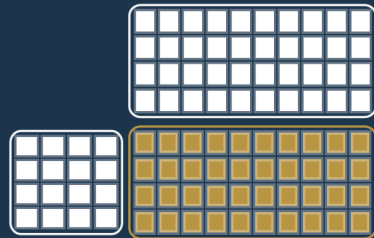
Reduction



Dot Product



Matrix Product

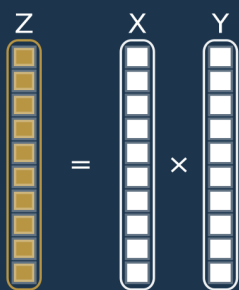


MKL



Simple Examples

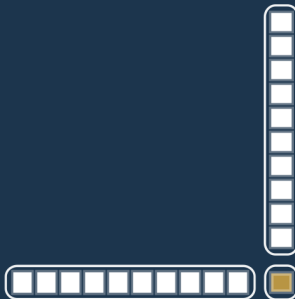
Hadamard Product



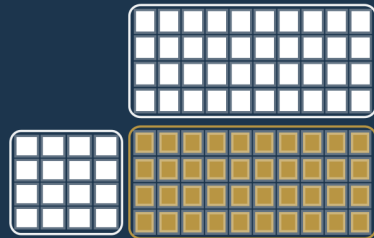
Reduction



Dot Product



Matrix Product



MKL

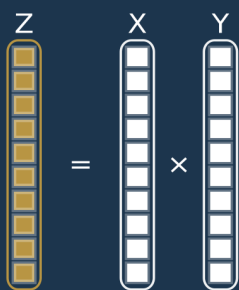


Eigen



Simple Examples

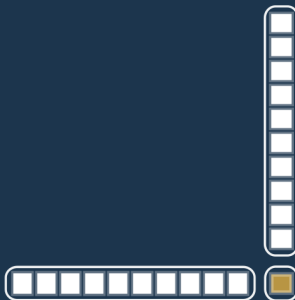
Hadamard Product



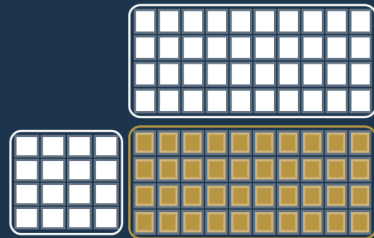
Reduction



Dot Product



Matrix Product



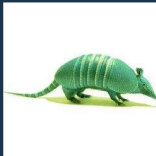
MKL



Eigen

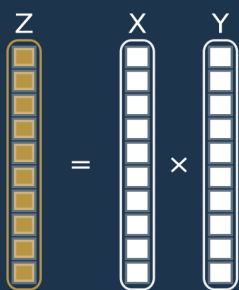


Armadillo



Simple Examples

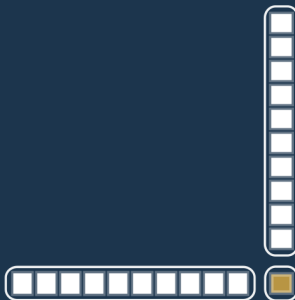
Hadamard Product



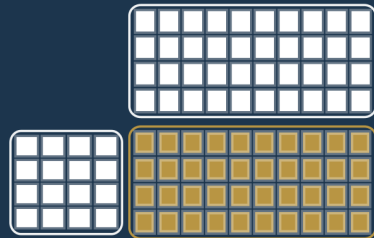
Reduction



Dot Product



Matrix Product



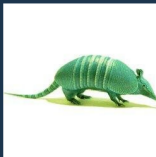
MKL



Eigen

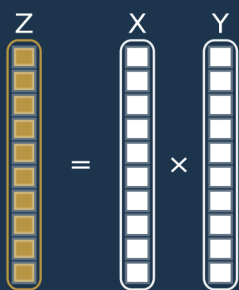


Armadillo



Simple Examples

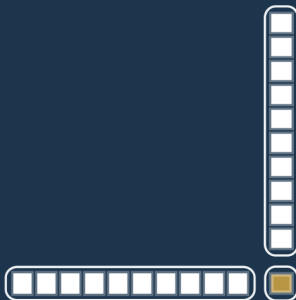
Hadamard Product



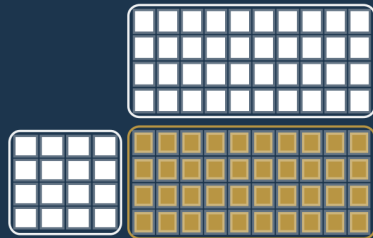
Reduction



Dot Product



Matrix Product



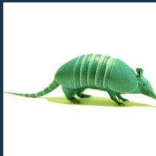
MKL



Eigen

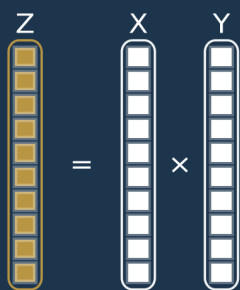


Armadillo



Simple Examples

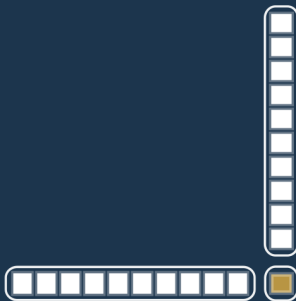
Hadamard Product



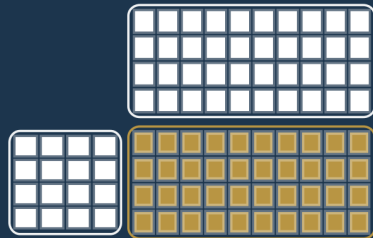
Reduction



Dot Product



Matrix Product



MKL



Eigen

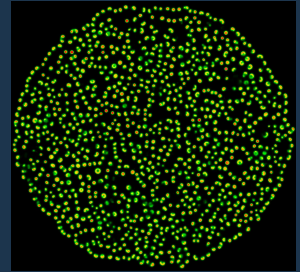


Armadillo

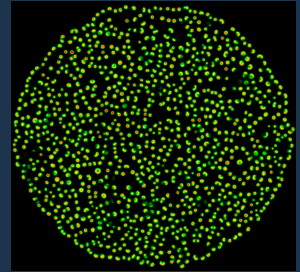


School computing example

Gray Scott reaction (a chemistry game of life)



Gray Scott reaction (a chemistry game of life)



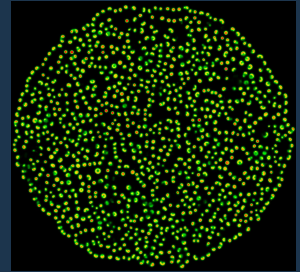
School computing example

Gray Scott reaction (a chemistry game of life)



Computing :

$$\begin{aligned} \frac{\partial u}{\partial t} &= r_u \nabla^2 u - uv^2 + f_r \times (1 - u) \\ \frac{\partial v}{\partial t} &= r_v \nabla^2 v + uv^2 - (f_r + k_r) \times v \end{aligned}$$



- ▶ u and v are concentration of product **U** and **V**
- ▶ r_u and r_v diffusion rate of **U** and **V**
- ▶ k_r (**Kill Rate**), conversion rate from **V** to **P**
- ▶ f_r (**Feed Rate**), speed of process which feed **U** and kills **V** and **P**
- ▶ $\nabla^2 u$ and $\nabla^2 v$ are différence of space concentration between current cell and its neighbours

School computing example

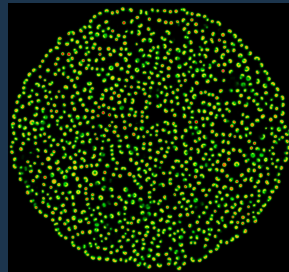
Gray Scott reaction (a chemistry game of life)



Computing :

$$\frac{\partial u}{\partial t} = r_u \nabla^2 u - uv^2 + f_r \times (1 - u)$$

$$\frac{\partial v}{\partial t} = r_v \nabla^2 v + uv^2 - (f_r + k_r) \times v$$



- ▶ u and v are concentration of product **U** and **V**
- ▶ r_u and r_v diffusion rate of **U** and **V**
- ▶ k_r (**Kill Rate**), conversion rate from **V** to **P**
- ▶ f_r (**Feed Rate**), speed of process which feed **U** and kills **V** and **P**
- ▶ $\nabla^2 u$ and $\nabla^2 v$ are différence of space concentration between current cell and its neighbours
- ▶ Easy to understand
- ▶ Not so easy for the compiler
- ▶ Possibility of high speed up





Fortran



Fortran





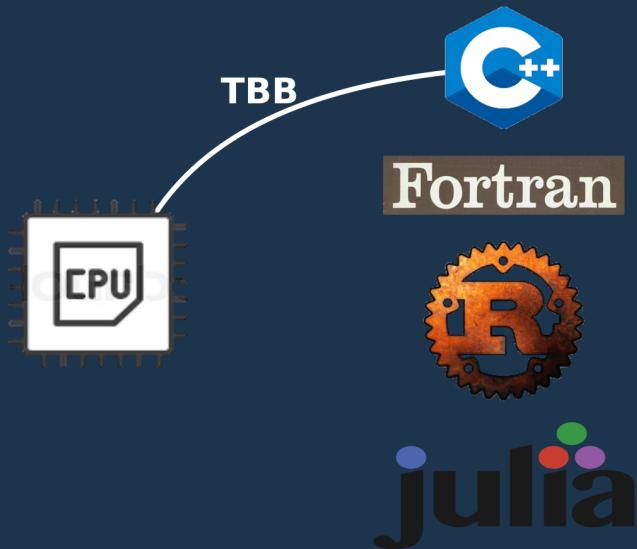
Fortran

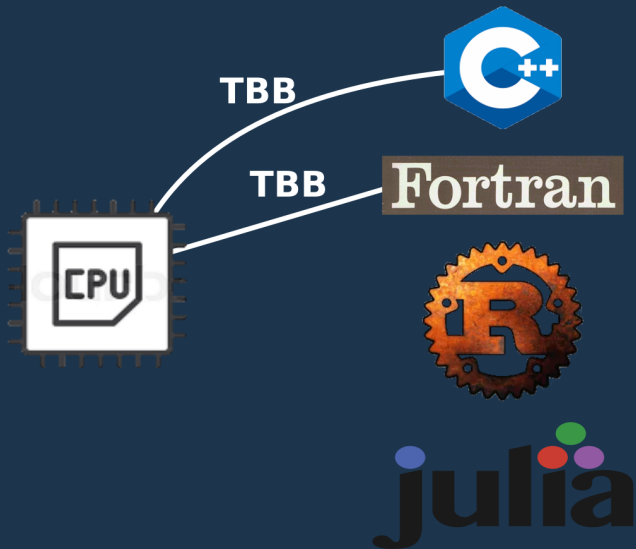


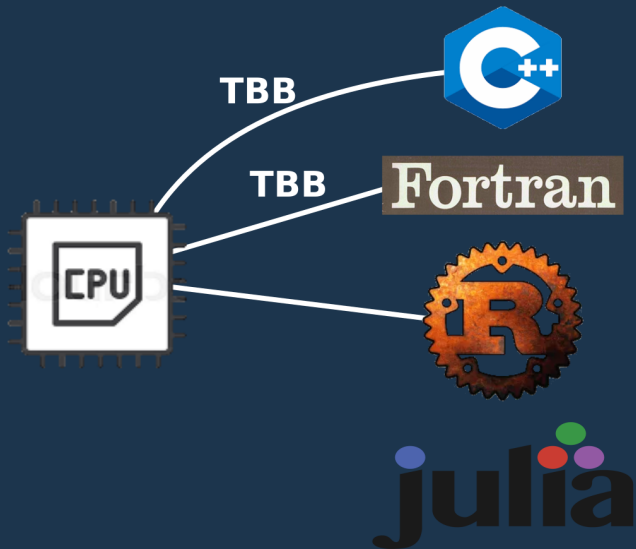


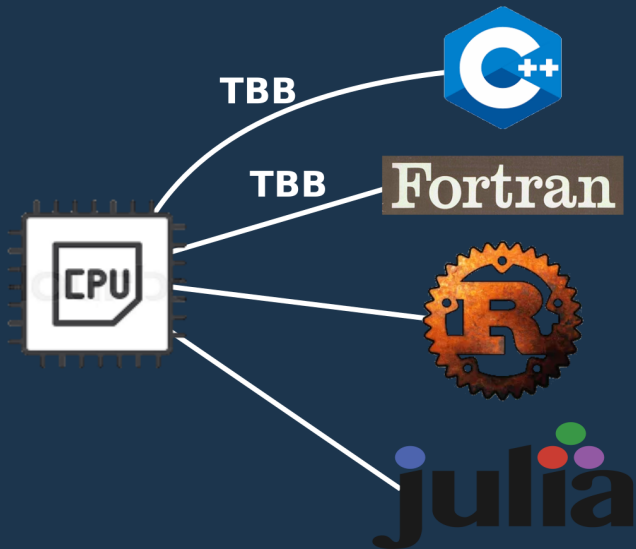
Fortran

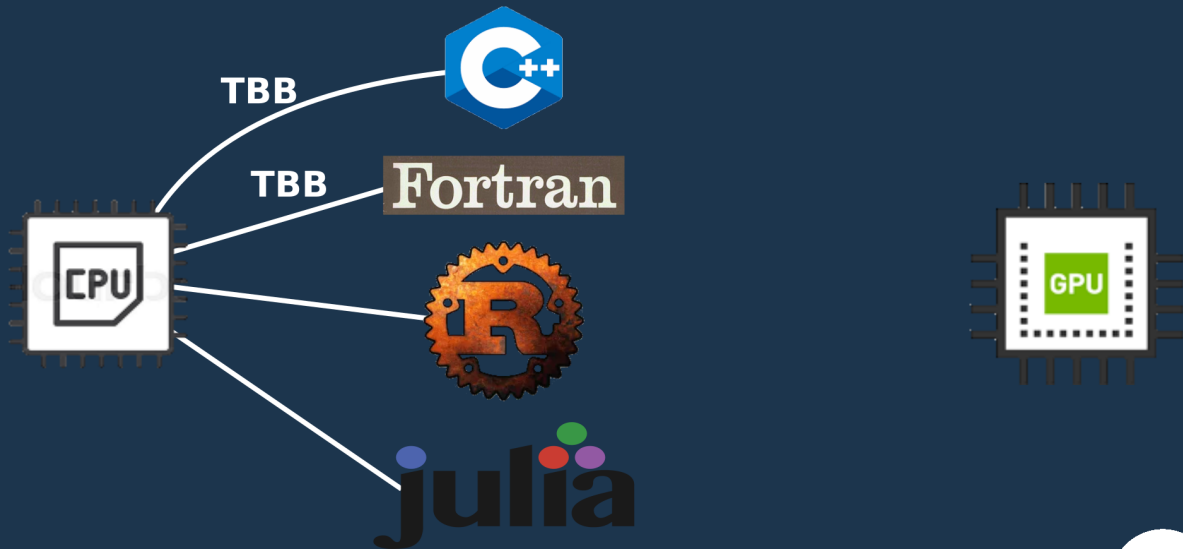


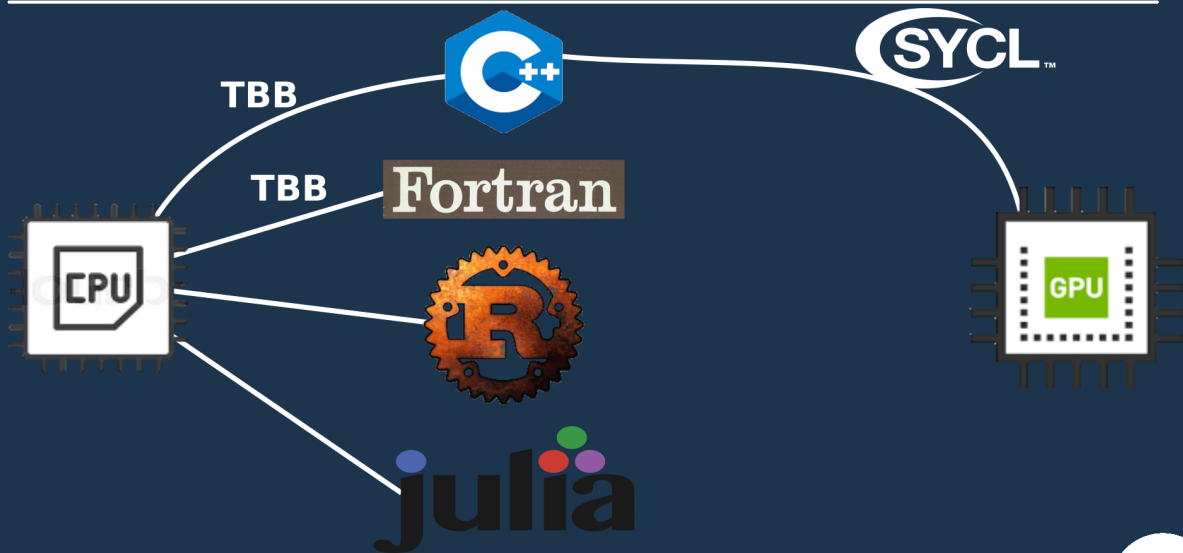


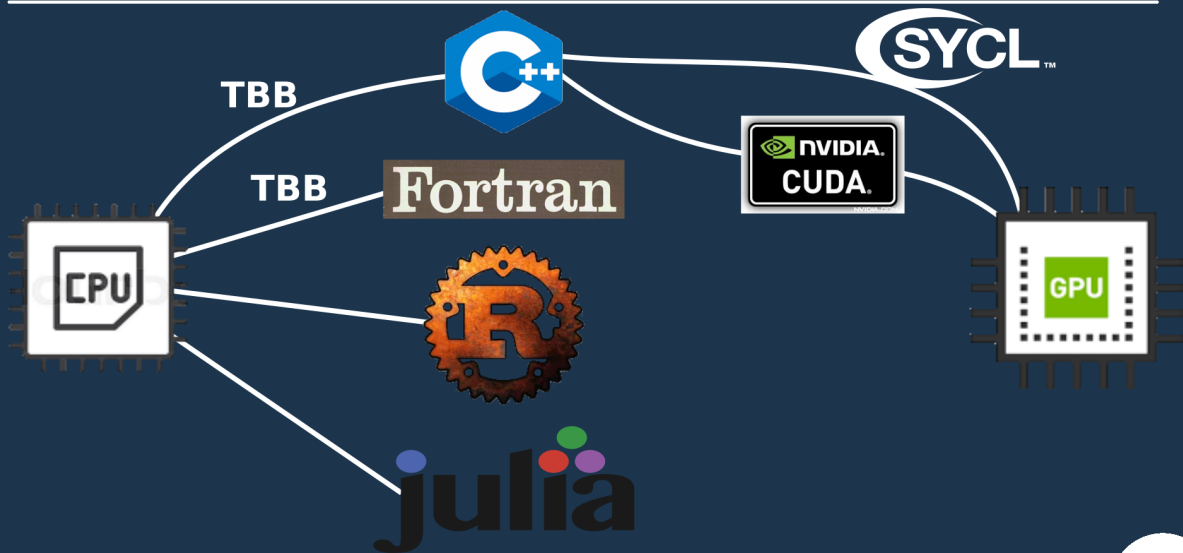


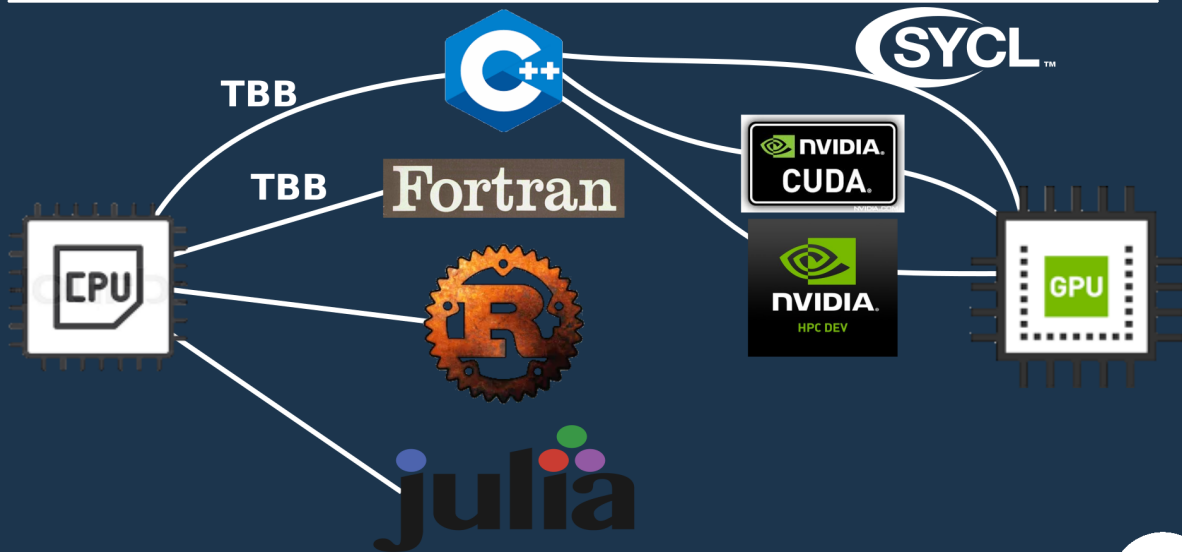


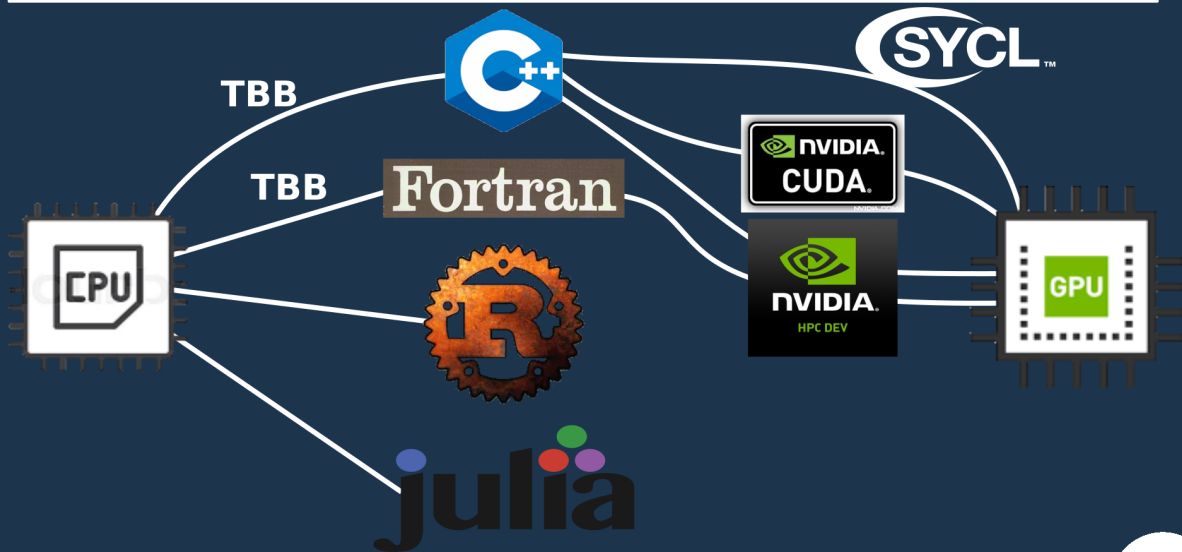


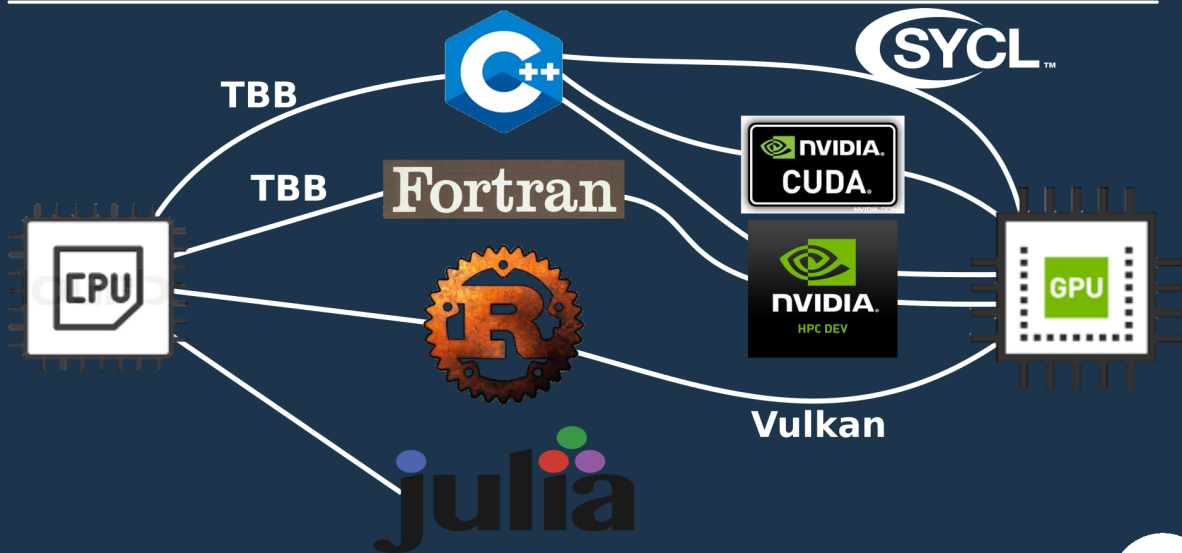


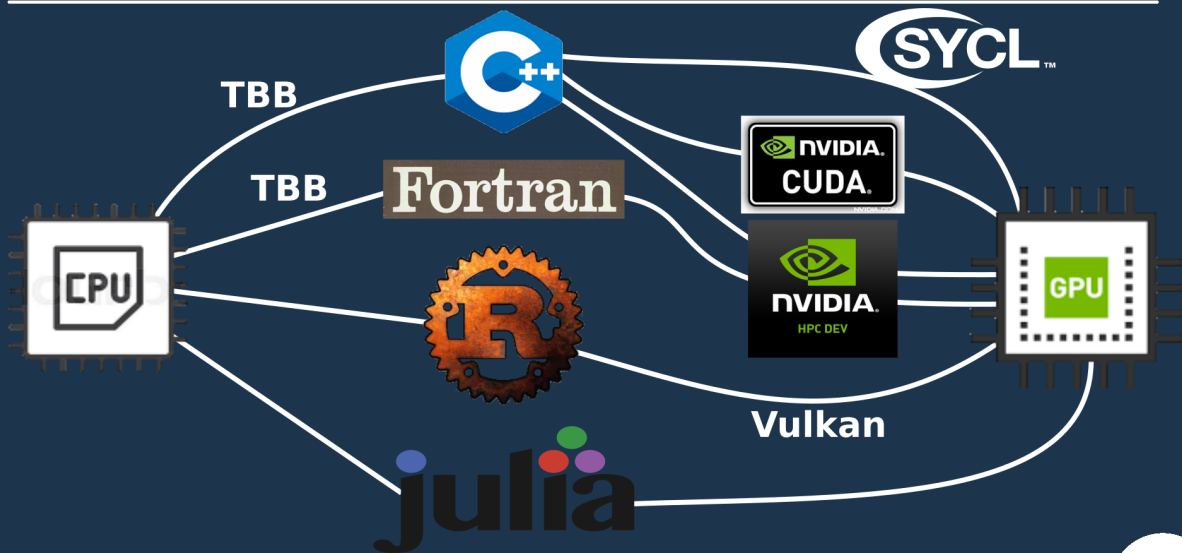










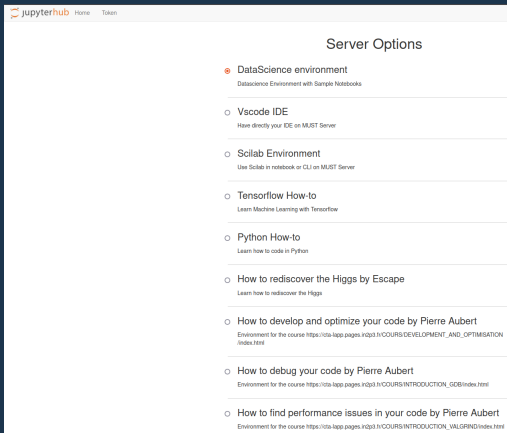


GPU MUST

	K80	P6000	T4	V100	A100	A100 80GB
TFlops (float)	8.73 (boost)	12.6	8.1	14	19.5	19.5
Memory (GB)	11.441 (24)	24	15	16	40	80
Nb Cuda Cores	2496 (4992)	3840	2560	5120	6912	2688
Clock rate (GHz)	0.824	1.645	1.590	1.380	1.410	1.410
Generation	3.7	6.1	7.5	7.0	8.0	8.0



<https://jupyter.must-dc.cloud/hub/>



The screenshot shows the JupyterHub interface with a list of server options. The first option, 'DataScience environment', is selected and highlighted with a red circle. The other options are listed below it, each with a radio button and a brief description.

- ☒ DataScience environment
DataScience Environment with Sample Notebooks
- ☐ Vscode IDE
Have directly your IDE on MUST Server
- ☐ Scilab Environment
Use Scilab in notebook or CLI on MUST Server
- ☐ Tensorflow How-to
Learn Machine Learning with Tensorflow
- ☐ Python How-to
Learn how to code in Python
- ☐ How to rediscover the Higgs by Escape
Learn how to rediscover the Higgs
- ☐ How to develop and optimize your code by Pierre Aubert
Environment for the course https://cta-lapp.pages.in2p3.fr/COURS/DEVELOPMENT_AND_OPTIMISATION/index.html
- ☐ How to debug your code by Pierre Aubert
Environment for the course https://cta-lapp.pages.in2p3.fr/COURS/INTRODUCTION_GDB/index.html
- ☐ How to find performance issues in your code by Pierre Aubert
Environment for the course https://cta-lapp.pages.in2p3.fr/COURS/INTRODUCTION_VALGRIND/index.html



HTCCondor
Software Suite

```
condor_submit job.submit
```

https://cta-lapp.pages.in2p3.fr/COURS/INTRODUCTION_MUST/

`https://indico.in2p3.fr/event/29755/timetable/#all.detailed`

`https://jupyter.must-dc.cloud/hub/`

`https://cta-lapp.pages.in2p3.fr/COURS/INTRODUCTION\_MUST/`