Future trends and challenges in scientific computing

Branislav Jansík

ACAT 2014

IT4Innovations
national01$#&0
supercomputing
center@#01%101

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND
INVESTING IN YOUR FUTURE

2007-13
OP Research and
Development for Innovation

Mission and Vision

Naše mise
Our mission is to deliver scientifically excellent and industry relevant research in the fields of high performance computing and embedded systems. We are providing state-of-the-art technology and expertise in high performance computing and embedded systems and make it available for Czech and international research teams from academia and industry.

Naše vize
To became top European Centre of Excellence in IT with the emphasis on high performance computing and embedded systems. With our research, know-how and infrastructure we aspire to improve the quality of life, to increase the competitiveness of industrial sector and to promote the cross-fertilization of high-performance computing, embedded systems and other scientific and technical disciplines.

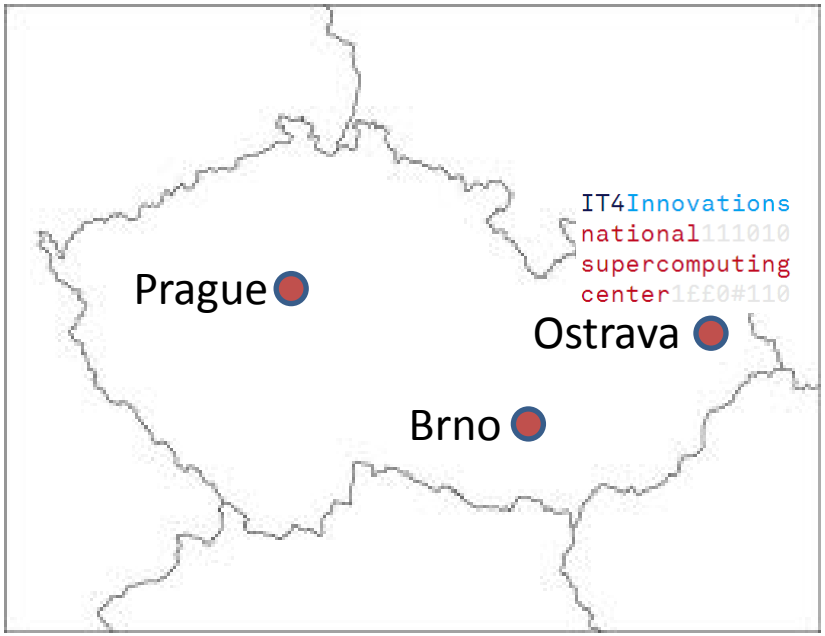
IT4Innovations
national01$#&0
supercomputing
center@#01%101

IT4Innovations
national01$#&0
supercomputing
center@#01%101

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND
INVESTING IN YOUR FUTURE

2007-13
OP Research and Development for Innovation

The IT4Innovations national supercomputing center



2013

2014

2015

94 TFLOPs system – most powerful supercomputer in Czech Republic, #6 in Central Europe (June 2013), operational since June 2013,
1 PFLOPs system planned for 2015

IT4Innovations
national
supercomputing
center

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND
INVESTING IN YOUR FUTURE

OP Research and Development for Innovation

Since 2010, IT4Innovations is representing Czech Republic within PRACE

PRACE Research Infrastructure
(Partnership for Advanced Computing in Europe)

http://www.prace-ri.eu

IT4Innovations:

- Helps to access EU HPC resources for researchers in Czech Republic (PRACE Wsh. Every ½ year)
- Contributes to EU HPC roadmap
- Commits to reserve 10% of IT4I resources (The DECI calls)

PRACE Spring school 2013 co-organizer (with Austria, Linz)

Anselm cluster in production since 1.6.2013

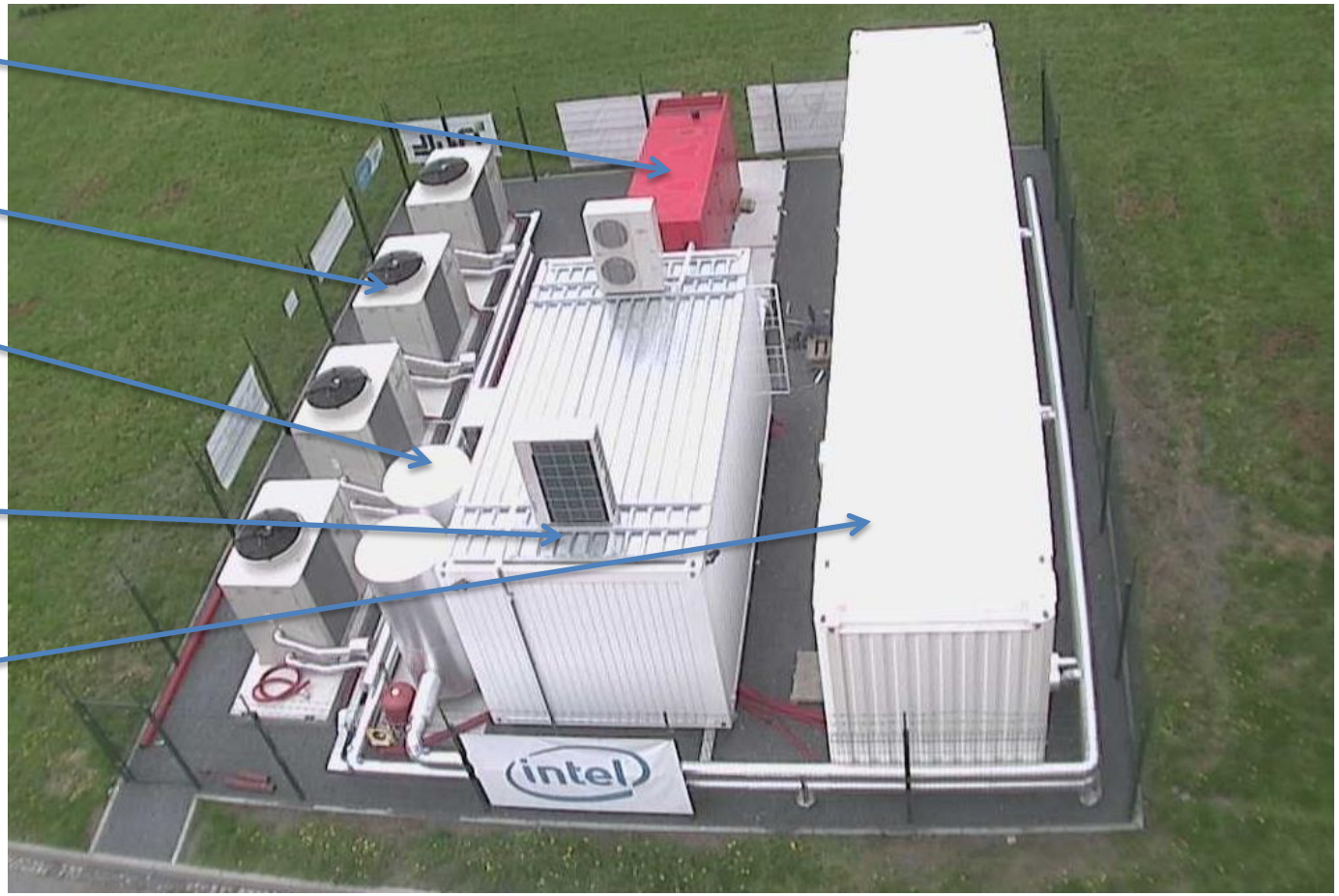
Diesel-generator

Four chillers

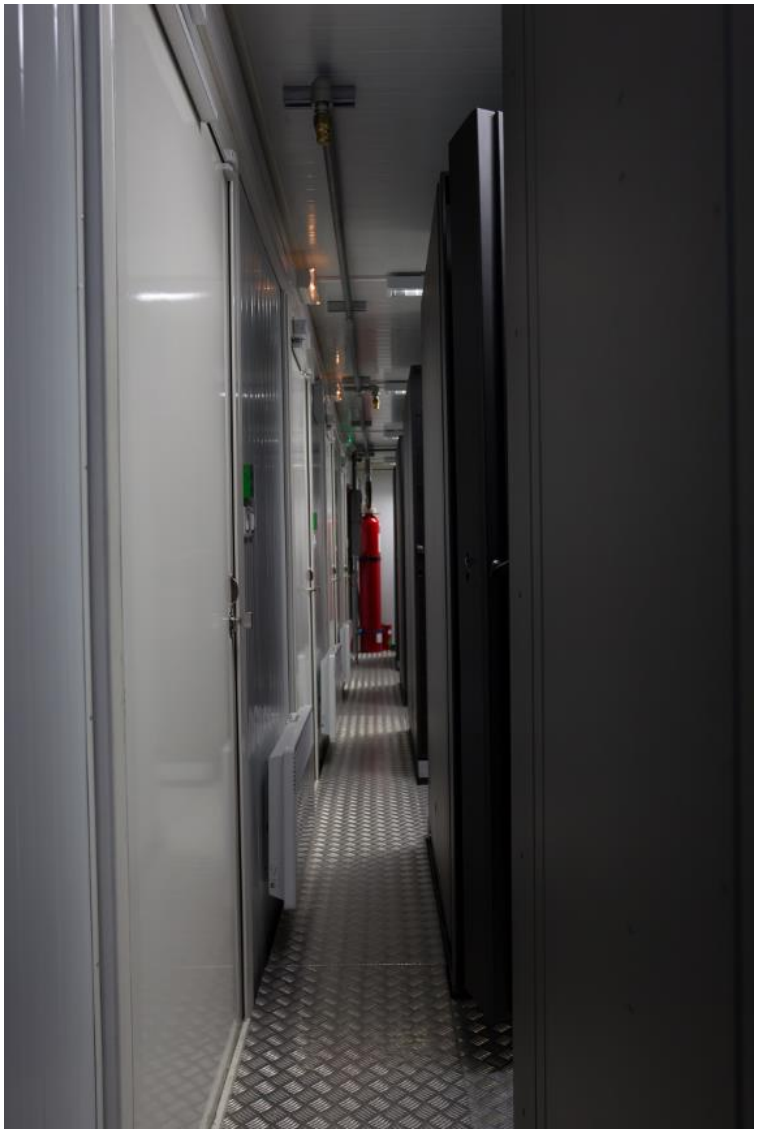
Cooling infrastructure

Service container

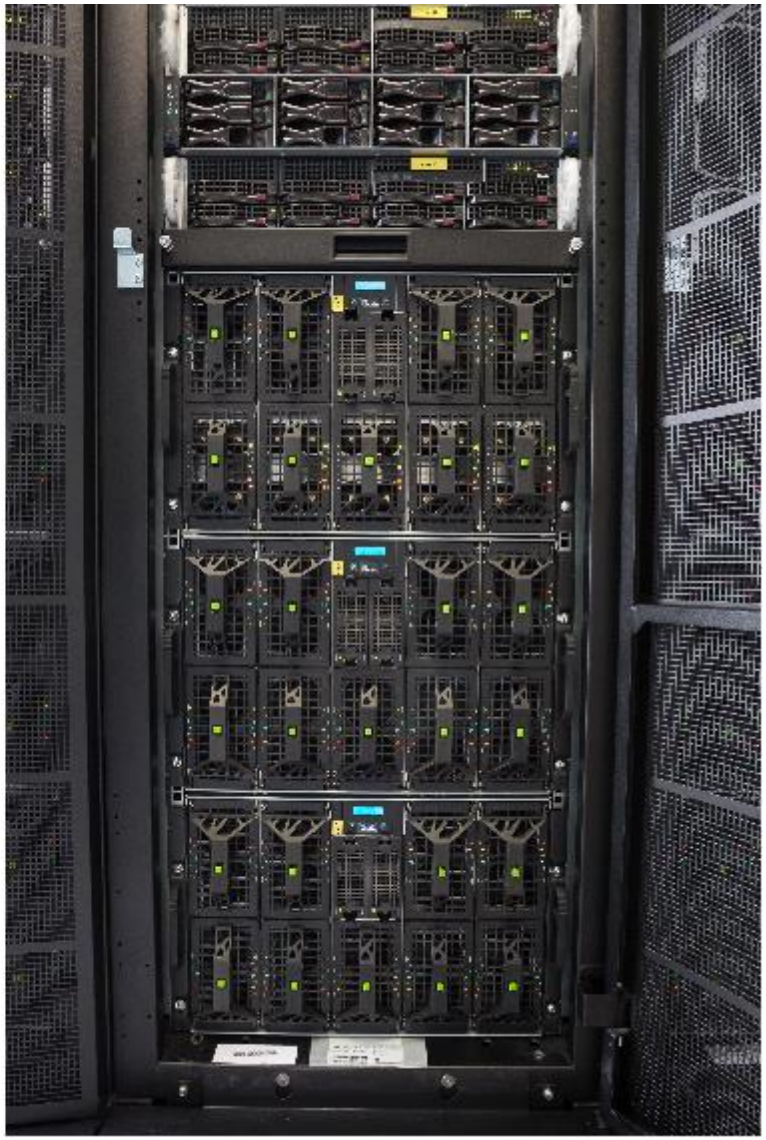
MOBULL container



What is inside



What is inside



What is inside

Computational system
90x bullx B510 double blades (180 nodes)
16 cores @2.4GHz, 64GB RAM, 500 GB HDD

27x bullx B515 blades
23x Nvidia Tesla K20
4x Intel Xeon Phi 5110P
16 cores @2.3GHz 64GB RAM 500 GB HDD

Network systems
40Gb/s Infiniband QDR, non blocking
8Gb/s FC SAN
1Gb/s LAN for administration
1Gb/s IPMI network

Backup system
Quantum Scalar i500
5TB/h backup rate

Storage systems
/home 300TB Lustre,
/scratch 135TB Lustre
2x bullx R423-E3 MDS, Netapp E2600 MDT
2x bullx R423-E3 OSS, Netapp E5400 OST
HA configuration

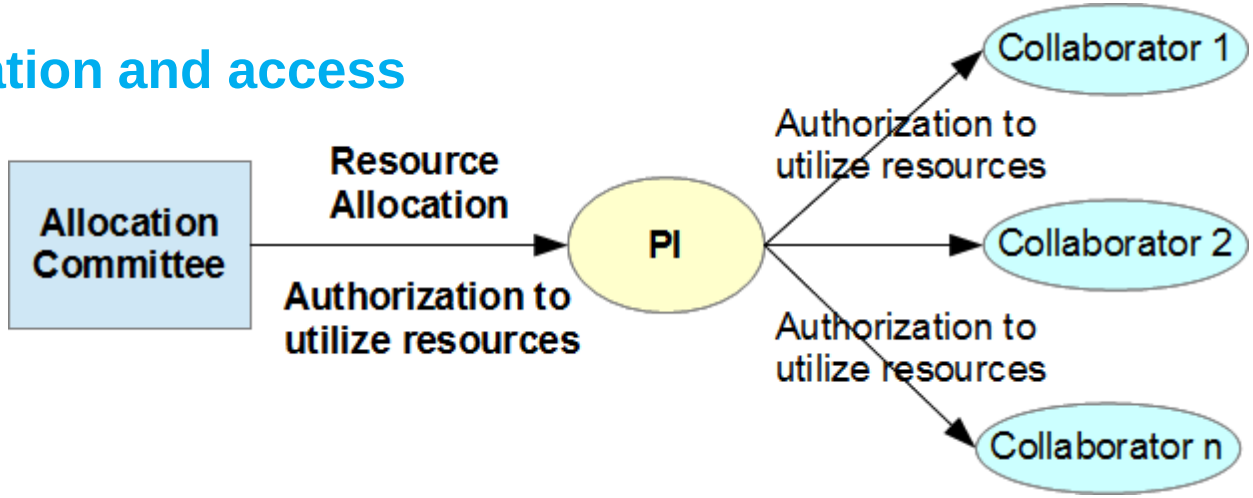
Virtualization system
vSphere software stack
3x bullx R423-E3, 128GB RAM
1x bullx R423-E3 control server
Dedicated Netapp E2600 block storage

Obtaining access to IT4Innovations

Submit a proposal

- IT4Innovations Open Access call for proposals, (2x a year)
  40% capacity, 4 750 000 core hours per call
- Directors discretinon access (any time)
  10% capacity

Authorization and access



Proposal content and evaluation

Aims and objectives Methods and state-of-the-art Impact and outlooks	Scientific readiness max 2 pages	1-5
Computational approach Parallelization and scalability Resources requested	Computational readiness max 1 pages	1-5
Socioeconomic impact Operational costs Amortization and renewal	Economic readiness	0-5

Parallel R

http://docs.it4i.cz/

Parallel execution of R may be achieved in many ways. One approach is the implied parallelization due to linked libraries or specially enabled functions, as described above. In the following sections, we focus on explicit parallelization, where parallel constructs are directly stated within the R script.

Package parallel

The package parallel provides support for parallel computation, including by forking (taken from package multicore), by sockets (taken from package snow) and random-number generation.

The package is activated this way:

```
$ R
> library(parallel)
```

More information and examples may be obtained directly by reading the documentation available in R

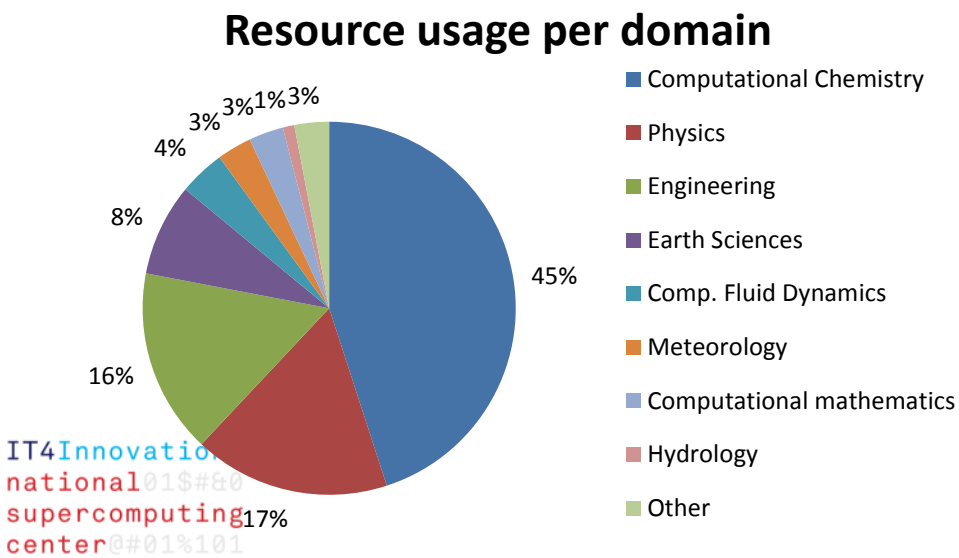
```
> ?parallel
> library(help = "parallel")
> vignette("parallel")
```

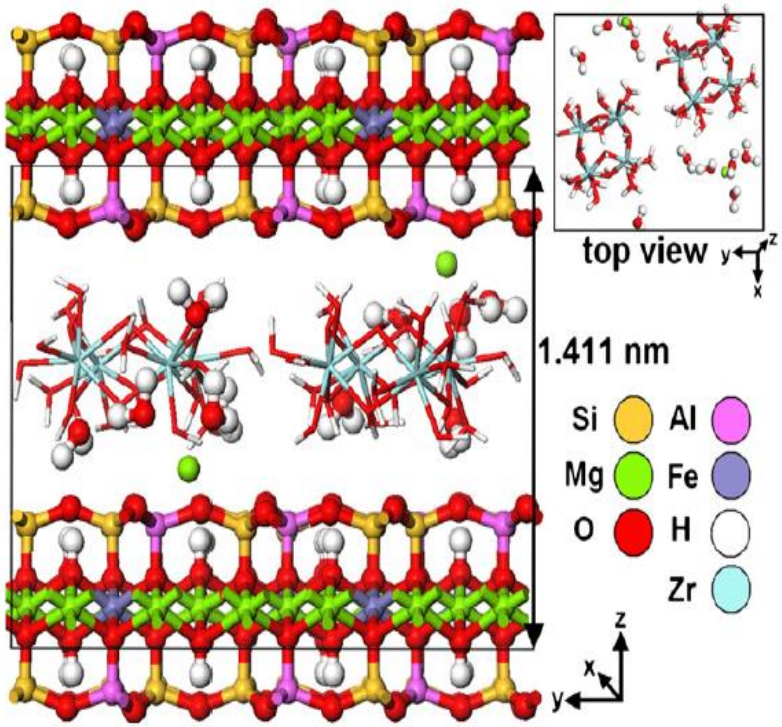
Download the package parallell vignette.

The forking is the most simple to use. Forking family of functions provide parallelized, drop in replacement for the serial apply() family of functions.

> Forking via package parallel provides functionality similar to OpenMP construct
> #omp parallel for

Anselm got #1 in Czech Republic

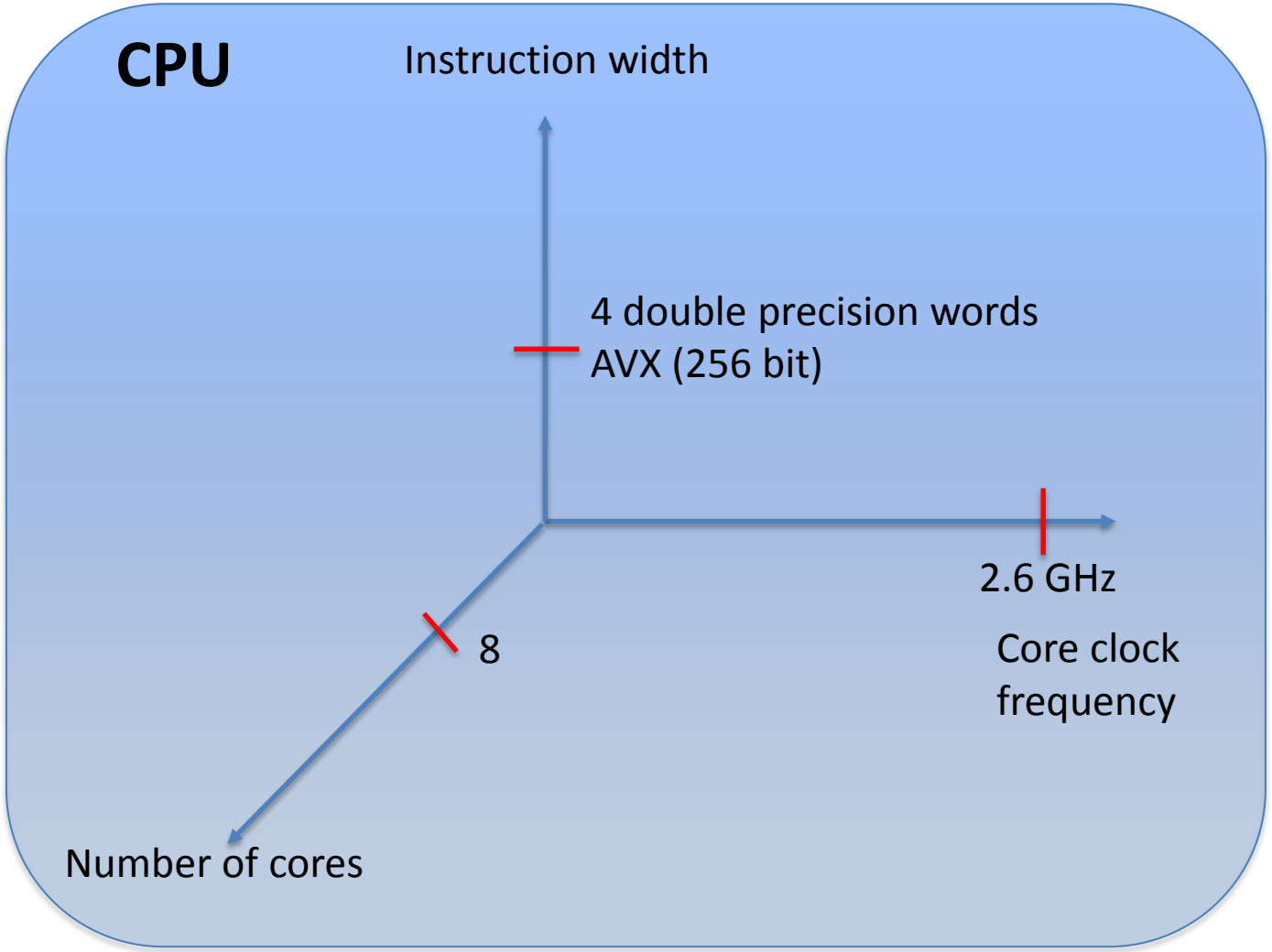




Two $[Zr_4(OH)_{14}(H2O)_{10}]^{2+}$ tetrameric cations, two Mg^{2+} cations and ten water molecules in the interlayer space of **vermiculite**.

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND
INVESTING IN YOUR FUTURE
OP Research and Development for Innovation

Challenges: CPUs and contemporary HPC



x 10^3

Peak CPU performance

$$R = F * O * N$$

R: Peak performance

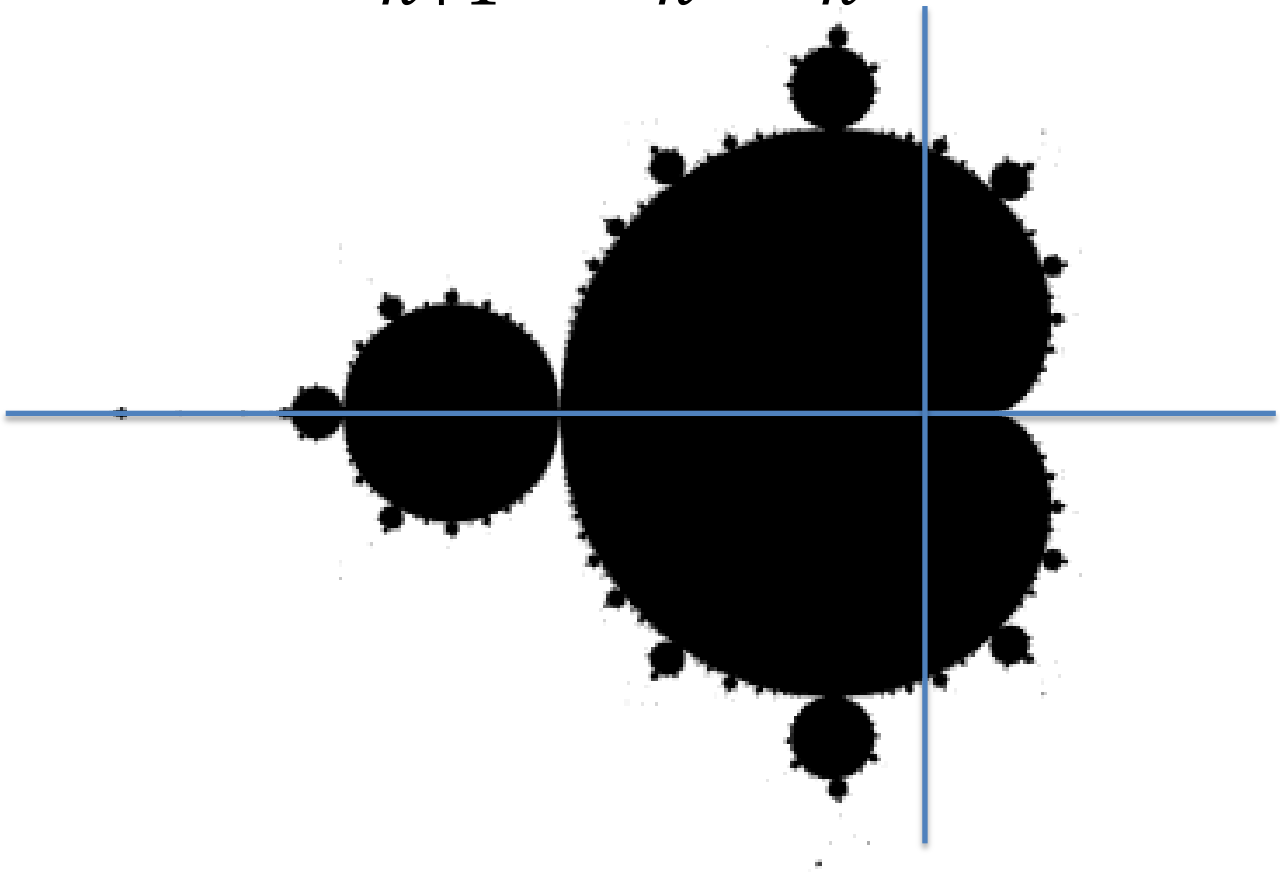
F: Core frequency

O: (FP) Operations per instruction

N: Number of cores

Mandelbrot set

$$z_{n+1} = z_n * z_n + c$$



Mandelbrot test

```
    "label:\n\t"
    "incq %%rcx\n\t"

REPEAT10(REPEAT10(
    "vmulpd %%ymm1, %%ymm1, %%ymm1\n\t"
    "vmulpd %%ymm3, %%ymm3, %%ymm3\n\t"
    "vmulpd %%ymm5, %%ymm5, %%ymm5\n\t"
    "vmulpd %%ymm7, %%ymm7, %%ymm7\n\t"
    "vmulpd %%ymm9, %%ymm9, %%ymm9\n\t"
    "vmulpd %%ymm11, %%ymm11, %%ymm11\n\t"
    "vmulpd %%ymm13, %%ymm13, %%ymm13\n\t"
    "vmulpd %%ymm15, %%ymm15, %%ymm15\n\t"
    "vaddpd %%ymm1, %%ymm0,  %%ymm1\n\t"
    "vaddpd %%ymm3, %%ymm2,  %%ymm3\n\t"
    "vaddpd %%ymm5, %%ymm4,  %%ymm5\n\t"
    "vaddpd %%ymm7, %%ymm6,  %%ymm7\n\t"
    "vaddpd %%ymm9, %%ymm8,  %%ymm9\n\t"
    "vaddpd %%ymm11, %%ymm10,  %%ymm11\n\t"
    "vaddpd %%ymm13, %%ymm12,  %%ymm13\n\t"
    "vaddpd %%ymm15, %%ymm14,  %%ymm15\n\t"
))
    "cmpq $20000000, %%rcx\n\t"
    "jb label \n\t"
```

1600 consecutive AVX Instructions, full exec pipeline

R

Mandelbrot test

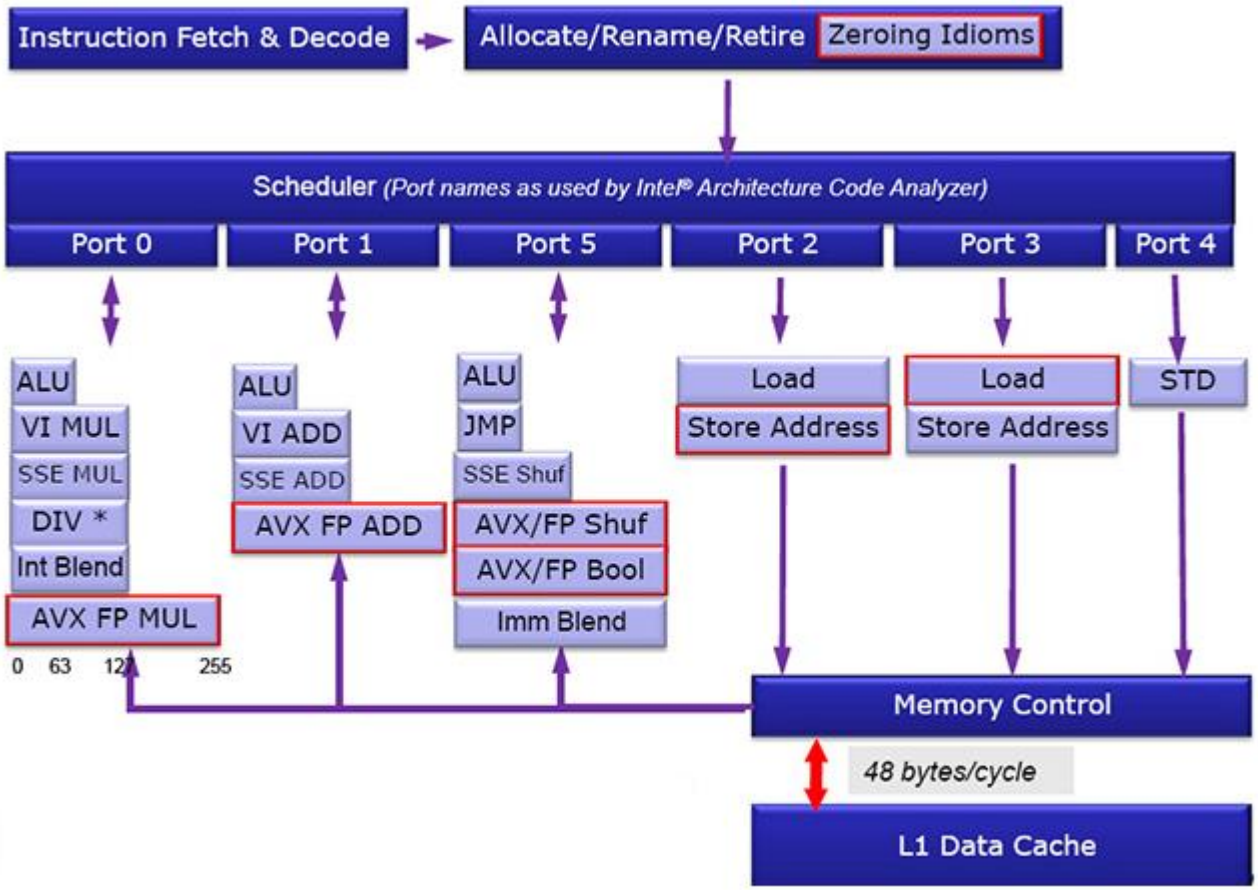
```
        "label:\n\t"
        "incq %%rcx\n\t"

    REPEAT10(REPEAT10(
        "vmulpd %%ymm1, %%ymm1, %%ymm1\n\t"
        "vmulpd %%ymm3, %%ymm3, %%ymm3\n\t"
        "vmulpd %%ymm5, %%ymm5, %%ymm5\n\t"
        "vmulpd %%ymm7, %%ymm7, %%ymm7\n\t"
```

Runs at 2x peak performance

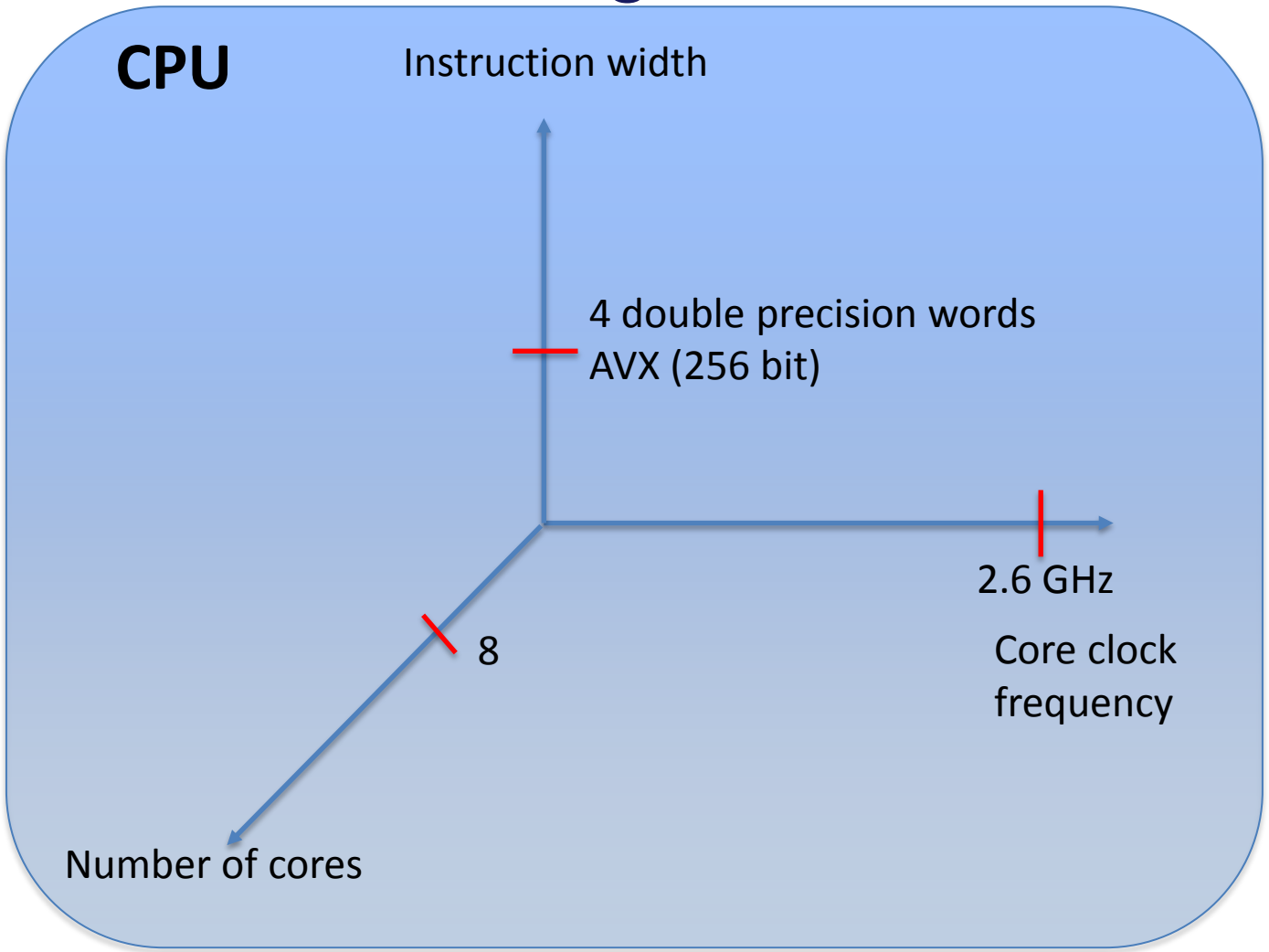
```
        "vaddpd %%ymm9, %%ymm8,  %%ymm9\n\t"
        "vaddpd %%ymm11, %%ymm10,  %%ymm11\n\t"
        "vaddpd %%ymm13, %%ymm12,  %%ymm13\n\t"
        "vaddpd %%ymm15, %%ymm14,  %%ymm15\n\t"
    ))
        "cmpq $20000000, %%rcx\n\t"
        "jb label \n\t"
```

R

Intel Sandybridge microarchitecture



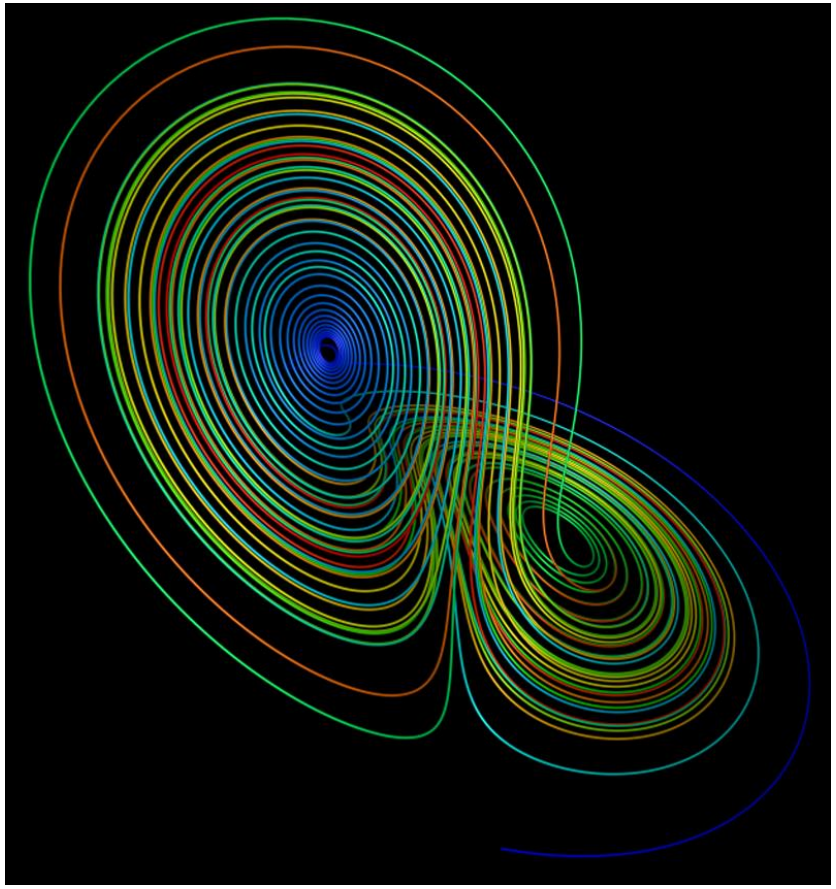
Challenges: Tools



x 10^3

Compiler challenges: Lorenz system

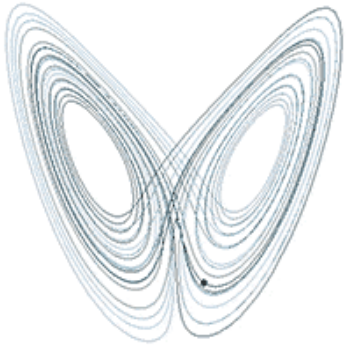
$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y - x),$$
$$\frac{\mathrm{d}y}{\mathrm{d}t} = x(\rho - z) - y,$$
$$\frac{\mathrm{d}z}{\mathrm{d}t} = xy - \beta z.$$



Compiler challenges: Lorenz system

C code

```
for (i=0; i<1000000000; i++) {
for (j=0; j<4; j++) {
 dx[j] = p[0]*(y[j]-x[j]);
 dy[j] = x[j]*(p[1]-z[j]) - y[j];
 dz[j] = x[j]*y[j]-p[2]*z[j];

 x[j] += p[3]*dx[j];
 y[j] += p[3]*dy[j];
 z[j] += p[3]*dz[j];
}
}
```



Hand optimized assembly code

```
        "label:\n\t"
        "incq %%rcx\n\t"

REPEAT10(
REPEAT10(
        "vsubpd %%ymm0, %%ymm1, %%ymm3\n\t"
        "vmulpd %%ymm3, %%ymm6, %%ymm3\n\t"
        "vsubpd %%ymm2, %%ymm7, %%ymm4\n\t"
        "vmulpd %%ymm4, %%ymm0, %%ymm4\n\t"
        "vsubpd %%ymm1, %%ymm4, %%ymm4\n\t"
        "vmulpd %%ymm0, %%ymm1, %%ymm10\n\t"
        "vmulpd %%ymm8, %%ymm2, %%ymm11\n\t"
        "vsubpd %%ymm11, %%ymm10, %%ymm5\n\t"
        "vmulpd %%ymm3, %%ymm9, %%ymm12\n\t"
        "vaddpd %%ymm0, %%ymm12, %%ymm0\n\t"
        "vmulpd %%ymm4, %%ymm9, %%ymm13\n\t"
        "vaddpd %%ymm1, %%ymm13, %%ymm1\n\t"
        "vmulpd %%ymm5, %%ymm9, %%ymm14\n\t"
        "vaddpd %%ymm2, %%ymm14, %%ymm2\n\t"
)
)
        "cmpq $10000000, %%rcx\n\t"
        "jb label \n\t"
```

Compiler challenges: Lorenz system

icc –S –O3 lorenz.c

```
..B1.3:                                  # Preds ..B1.3 ..B1.2
        movaps    128(%rsp,%rcx,8), %xmm9
        movaps    %xmm1, %xmm6
        subpd     %xmm9, %xmm6
        movaps    96(%rsp,%rcx,8), %xmm7
        movaps    64(%rsp,%rcx,8), %xmm5
        movaps    %xmm7, %xmm4
        movaps    .L_2il0floatpacket.6(%rip), %xmm3
        movaps    %xmm5, %xmm8
        subpd     %xmm5, %xmm4
        mulpd     %xmm5, %xmm6
        mulpd     %xmm7, %xmm8
        mulpd     %xmm9, %xmm3
        mulpd     %xmm2, %xmm4
        subpd     %xmm7, %xmm6
        subpd     %xmm3, %xmm8
        mulpd     %xmm0, %xmm4
        mulpd     %xmm0, %xmm6
        mulpd     %xmm0, %xmm8
        addpd     %xmm4, %xmm5
        addpd     %xmm6, %xmm7
        addpd     %xmm8, %xmm9
        movaps    %xmm5, 64(%rsp,%rcx,8)
        movaps    %xmm7, 96(%rsp,%rcx,8)
        movaps    %xmm9, 128(%rsp,%rcx,8)
        addq      $2, %rcx
        cmpq      $4, %rcx
        jb        ..B1.3          # Prob 75%
```

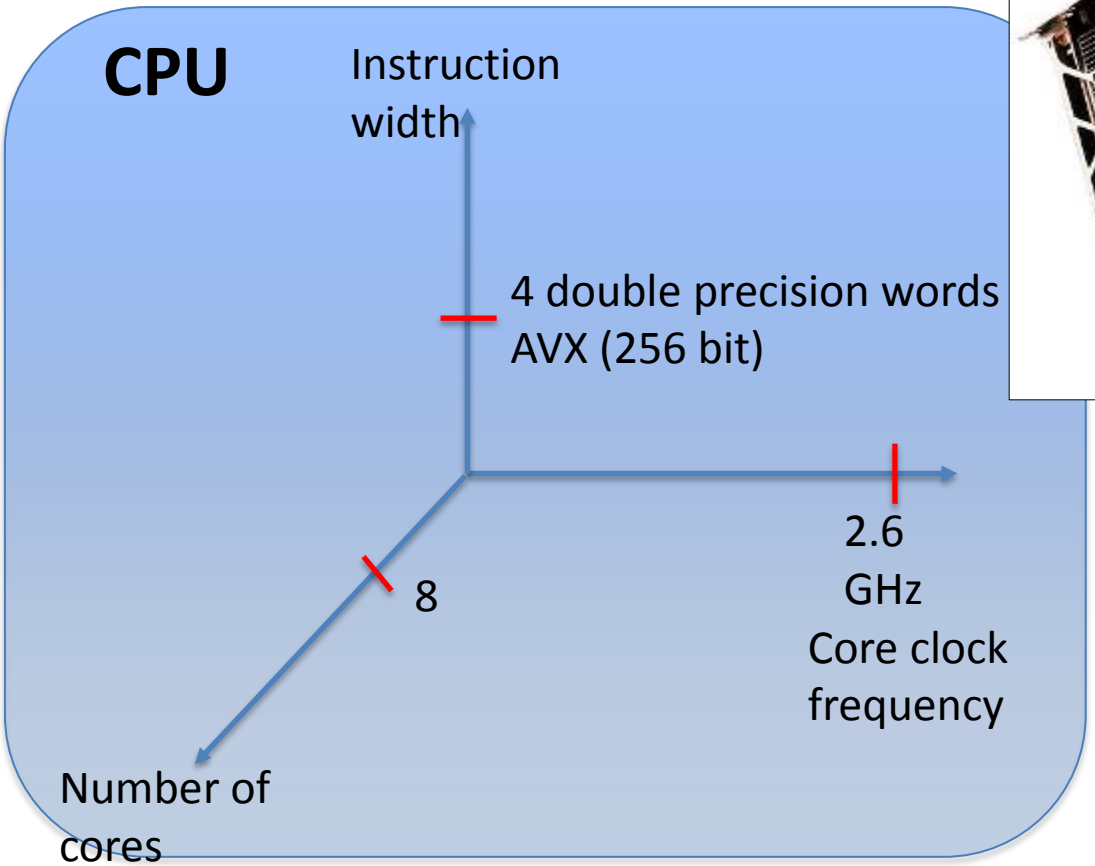
Runtime 12 secs

icc –S –O3 –xavx lorenz.c

```
..B1.2:                                  # Preds ..B1.2 ..B1.10
        vsubpd    %ymm5, %ymm4, %ymm8
        incq      %rax
        vsubpd    %ymm1, %ymm6, %ymm7
        vmulpd    %ymm6, %ymm1, %ymm10
        vmulpd    %ymm8, %ymm1, %ymm9
        vmulpd    %ymm5, %ymm0, %ymm11
        vmulpd    %ymm7, %ymm2, %ymm12
        vsubpd    %ymm6, %ymm9, %ymm14
        vsubpd    %ymm11, %ymm10, %ymm7
        vmulpd    %ymm12, %ymm3, %ymm13
        vmulpd    %ymm14, %ymm3, %ymm15
        vmulpd    %ymm7, %ymm3, %ymm8
        vaddpd    %ymm13, %ymm1, %ymm1
        vaddpd    %ymm15, %ymm6, %ymm6
        vaddpd    %ymm8, %ymm5, %ymm5
        cmpq      $1000000000, %rax
        jb        ..B1.2          # Prob 99%
```

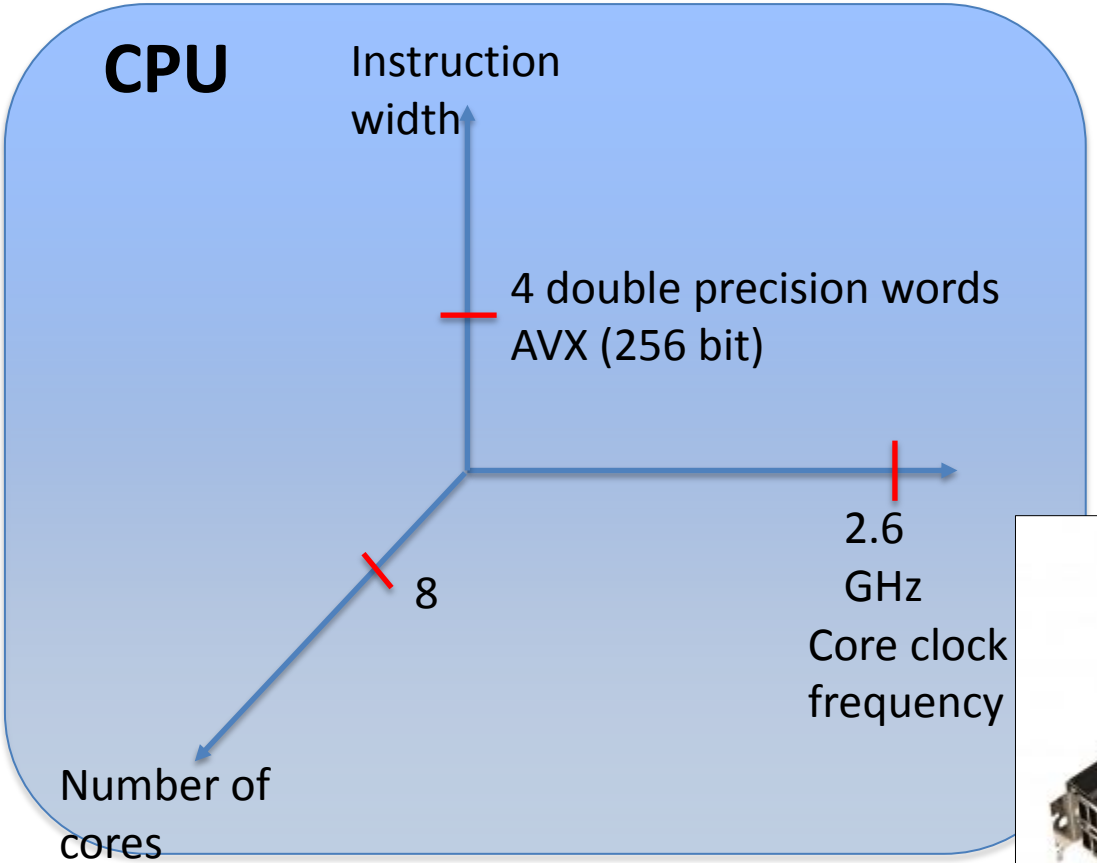
Runtime 7 secs

Challenges: Accelerated architectures



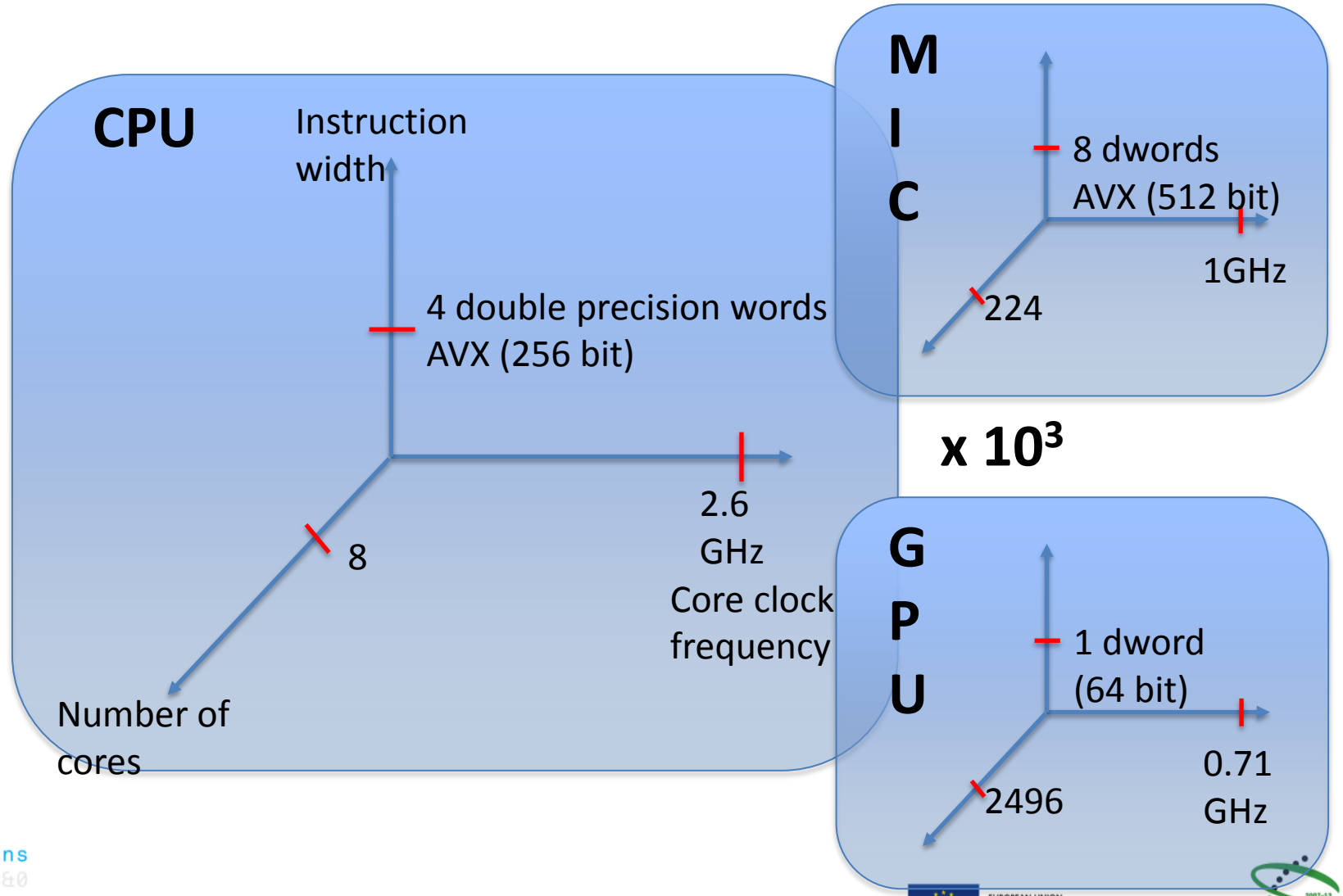
x 10^3

Challenges: Accelerated architectures

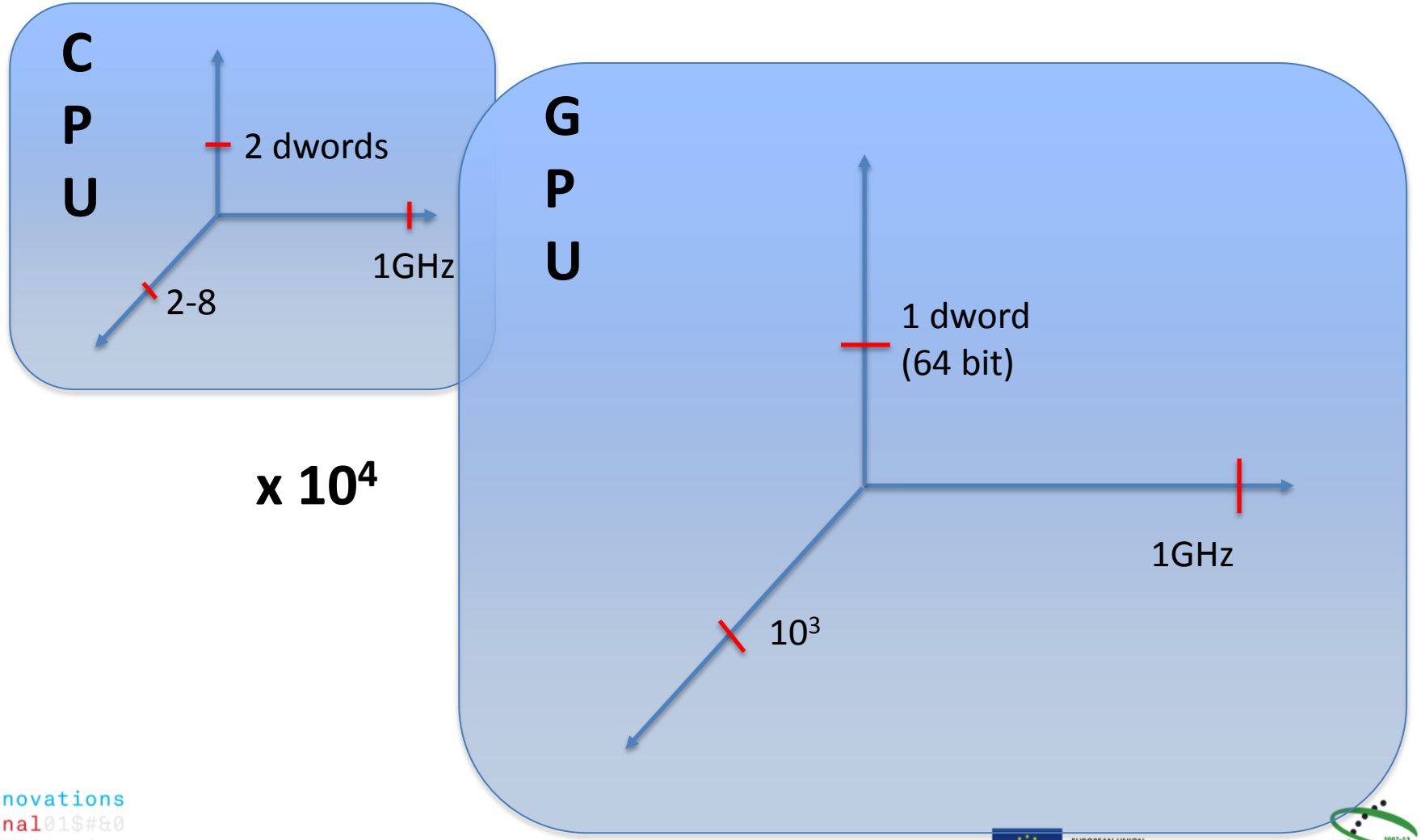


x 10^3

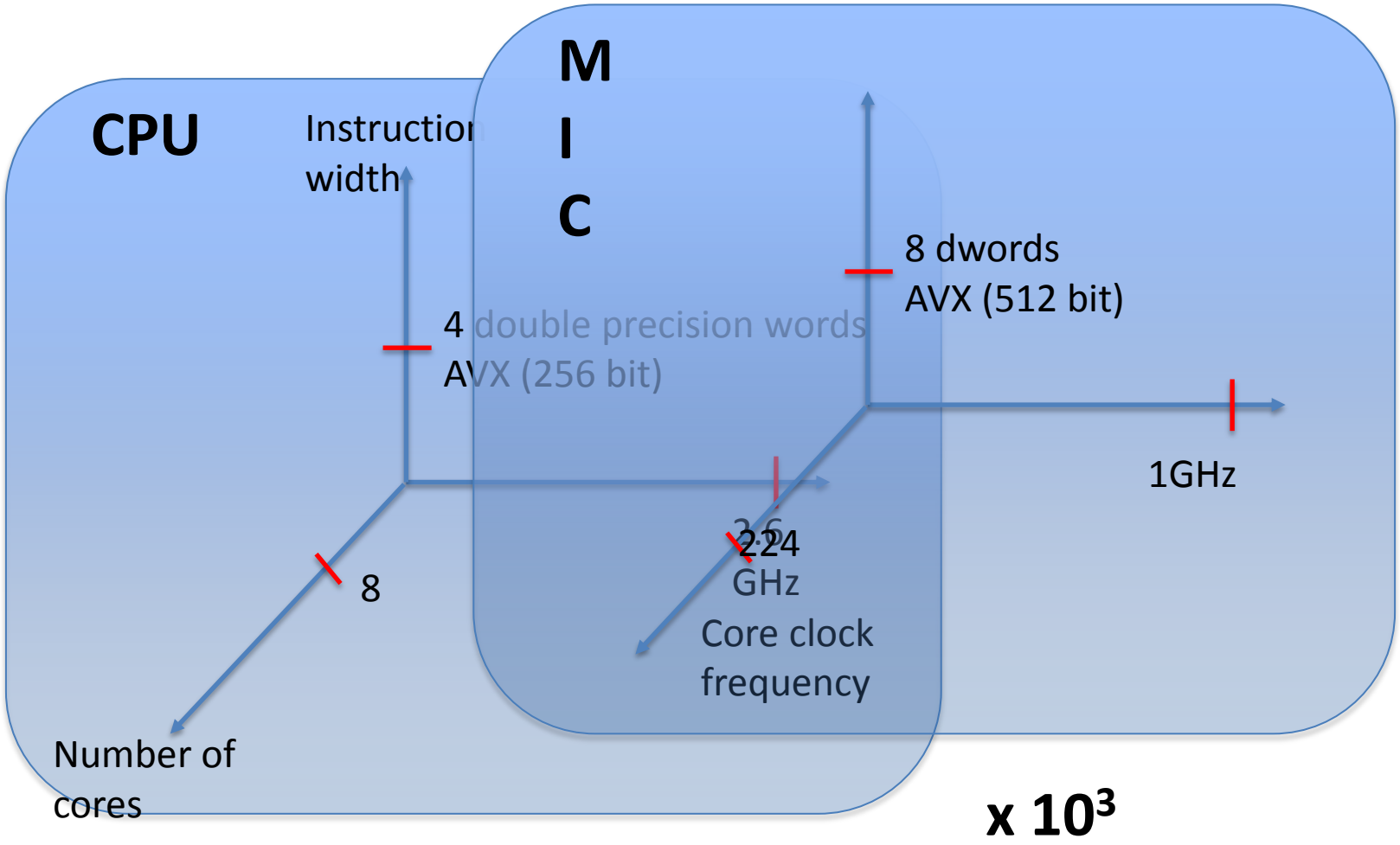
Challenges: Accelerated architectures



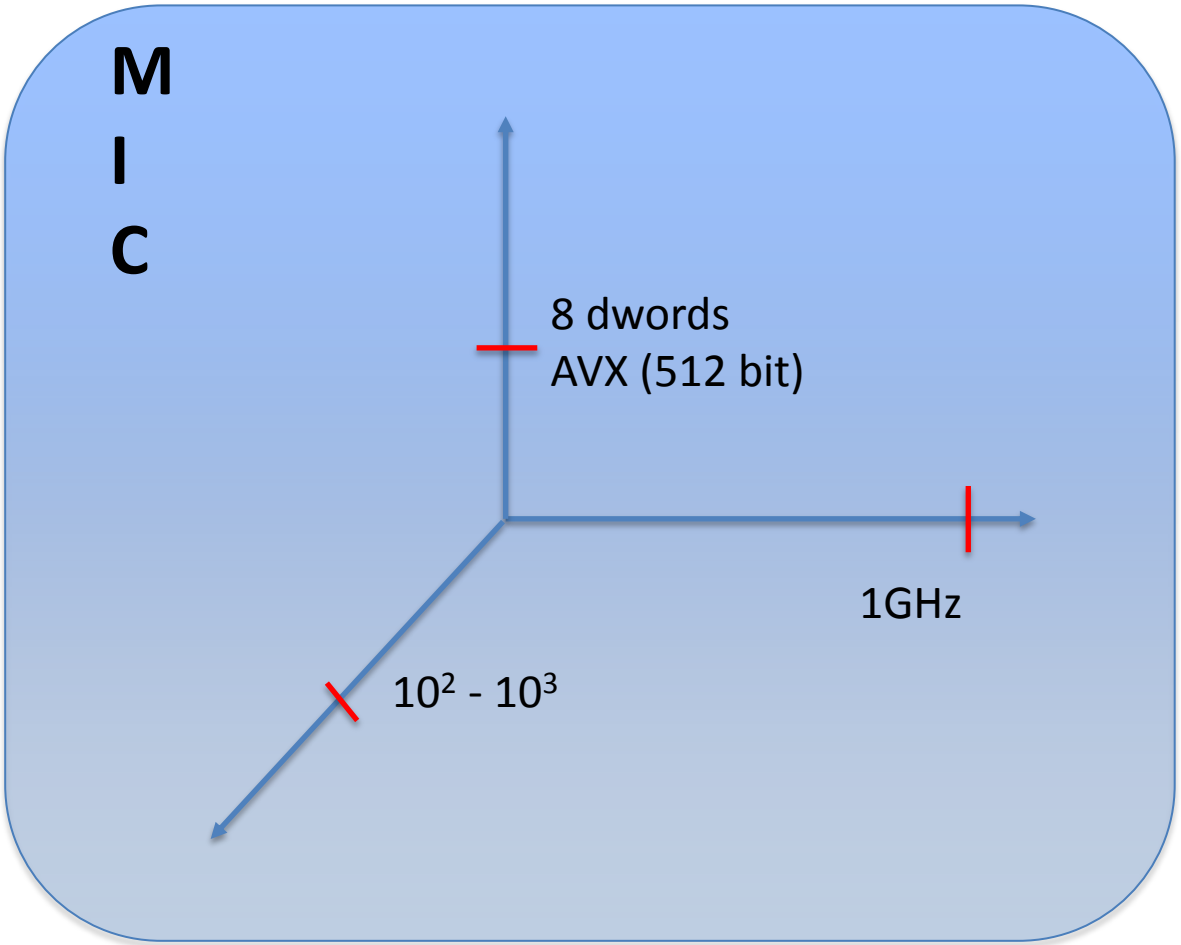
Future trends: small cores with a lot of GPU acceleration



Future trends: Many integrated cores

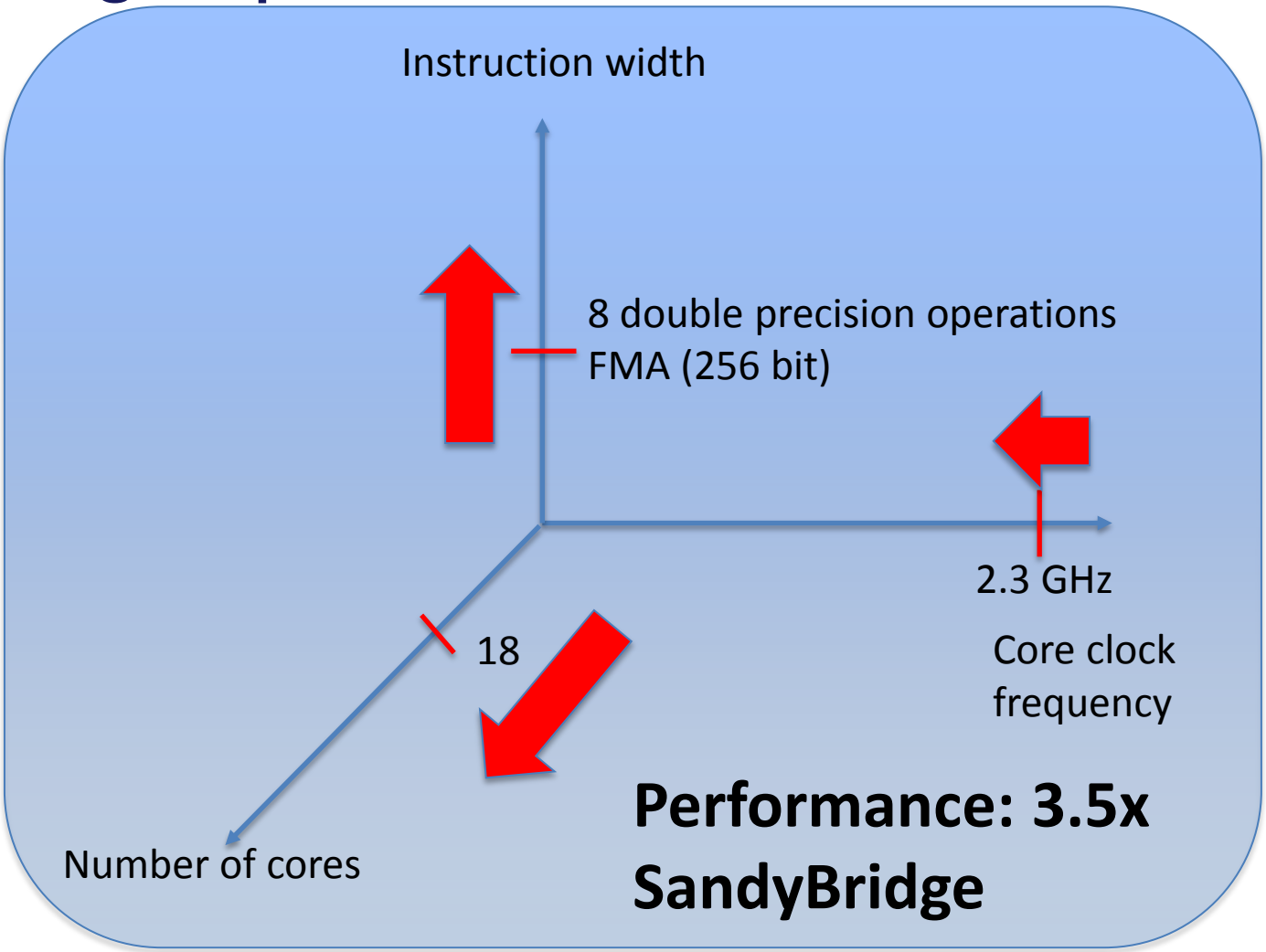


Future trends: Many integrated cores



x 10^3

A glimpse of future: Intel Haswell



Conclusions

- IT4Innovations: resources, expertise and collaboration in HPC domain
- Efficient HPC – a 4D vector space
- Unique challenges in every dimension, tools not always adequate
- Future: The space envelope is changing: a lot more of weak cores

branislav.jansik@vsb.cz