EasyBuild @ HPCNow!

Oriol Mula Valls (HPCNow!)

Agenda

1. Introduction
2. HPCNow!
3. EasyBuild

Who I am

- BSc @ UAB
- MSc @ UPC
- @omulavalls
- @omula
- Learnt about EasyBuild in HPCKP'14.

Agenda

1. Introduction
2. HPCNow!
   - Who
   - What
   - Where
3. EasyBuild

Who

Who we are?

- Created on Oct 2012
- $\approx$ 15 people
- $>$ 50 customers
- $>$ 80 projects
- $>$ 25 EasyBuild deployments

What

What we do?

- Installation (hardware and software)
- Training
- Support
- Managed services
- Consultancy

Where

Apart from Spain and New Zealand

- Bangladesh
- Germany
- Mexico
- Morocco
- USA
- ...

Agenda

1. Introduction
2. HPCNow!
3. EasyBuild

Challenges and questions

- Do we have root access? OS installed?
- Where can we put the software? Which filesystem do we have to rely on? (e.g.: Perl compilation fails on BeeGFS)
- How many architectures and/or μarchitectures? Mixed environment?
- Will the cluster grow? (Newer μarchitectures?)
- What about `--robot`?
- Where should we put a Python package? Inside Python module and use `--skip --rebuild`? Shall we create PyhonBundle or PythonPackage?

EasyBuild usage I

Software building variables

```
...
export EASYBUILD_PREFIX=${SOFTWARE_ROOT}/${OS_NAME}/${OS_VERSION}/${ARCHITECTURE}
export EASYBUILD_BUILDPATH=/dev/shm
...
```

/etc/profile.d/hpcnow-easybuild.sh
Without root access we put the file in another directoy and source it in ~/.bashrc

${SOFTWARE_ROOT}/CentOS/7.6.1810/Skylake/
${SOFTWARE_ROOT}/common (for binaries)

EasyBuild usage II

We define

- **$OS** `lsb_release -si`
  - CentOS
  - openSUSE
  - RHEL
  - SUSE
- **$OSVERSION** `lsb_release -sr`
- **$ARCHITECTURE** `${cpu_family}${cpu_model}`
  - **CPU Family**
    `lscpu | grep "CPU family:" | awk '{printf "%02X",$3}'`
  - **CPU Model**
    `lscpu | grep "Model:" | awk '{printf "%02X",$2}'`

AVX512

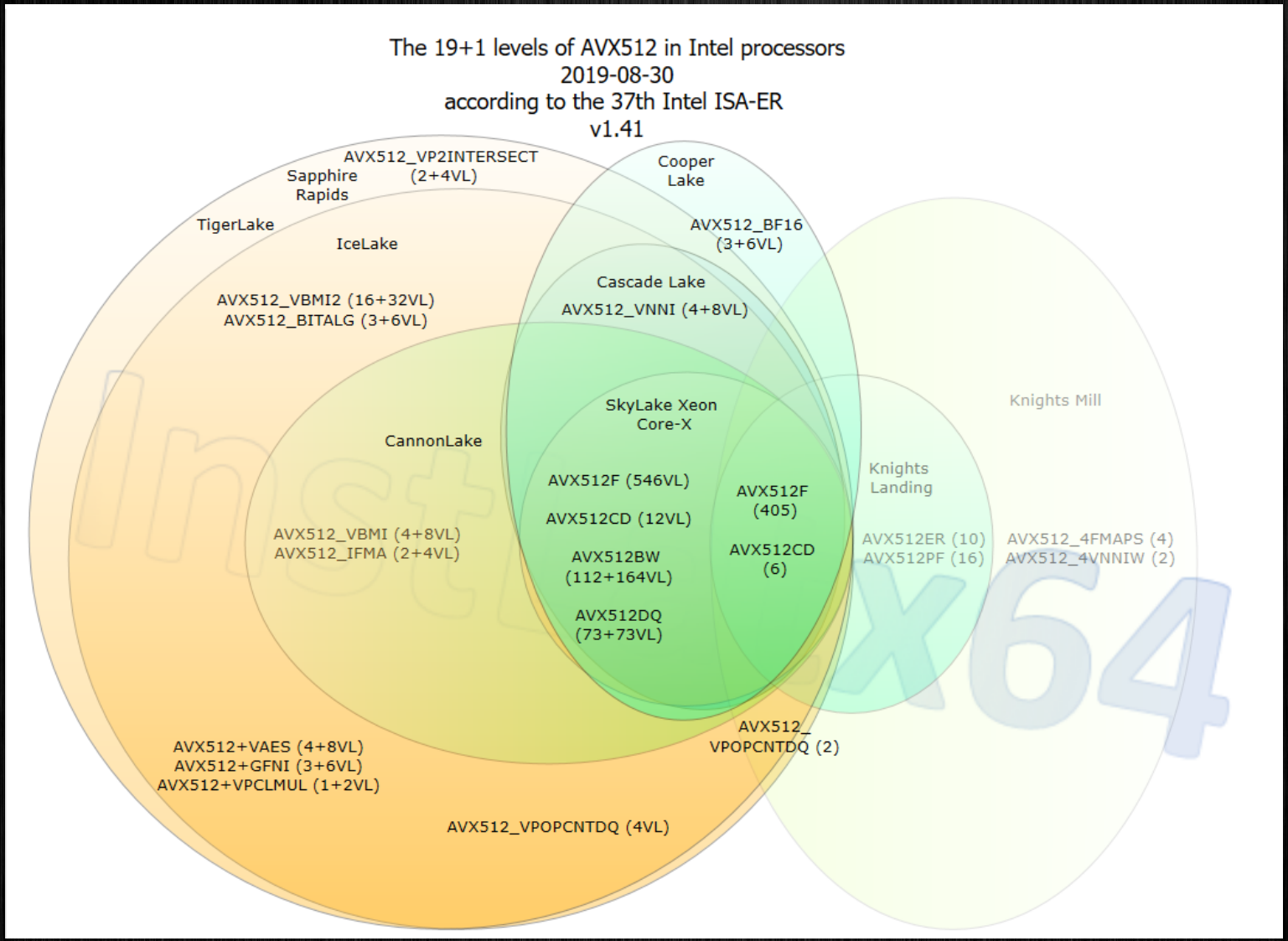

Figure: Source: http://instlatx64.atw.hu/

Core	Extended Family	Family	Extended Model	Model	Stepping
Skylake (X, SP, DE, W)	0	0x6	0x5	0x5	
Cascade Lake	0	0x6	0x5	0x5	0x5..0x7
Zen (Napples)	0x8	0xF	0x0..0x1	0x1	

Table: Source: https://en.wikichip.org

Core	Extended Family	Family	Extended Model	Model	Stepping
Skylake (X, SP, DE, W)	0	0x6	0x5	0x5	
Cascade Lake	0	0x6	0x5	0x5	0x5..0x7
Zen (Napples)	0x8	0xF	0x0..0x1	0x1	

Table: Source: https://en.wikichip.org

Compiler instruction scheduling

Different architectures

Different architectures

```
$ lscpu
Architecture:          ppc64le
Byte Order:            Little Endian
CPU(s):                160
On-line CPU(s) list:   0-159
Thread(s) per core:    4
Core(s) per socket:    20
Socket(s):             2
NUMA node(s):          6
Model:                 2.1 (pvr 004e 1201)
Model name:            POWER9, altivec supported
CPU max MHz:           3800.0000
CPU min MHz:           2300.0000
L1d cache:             32K
L1i cache:             32K
L2 cache:              512K
L3 cache:              10240K
NUMA node0 CPU(s):     0-79
NUMA node8 CPU(s):     80-159
NUMA node252 CPU(s):
NUMA node253 CPU(s):
NUMA node254 CPU(s):
NUMA node255 CPU(s):
```

```
$ cat /proc/cpuinfo
processor       : 0
cpu             : POWER9, altivec supported
clock           : 3616.000000MHz
revision        : 2.1 (pvr 004e 1201)

...

processor       : 159
cpu             : POWER9, altivec supported
clock           : 3616.000000MHz
revision        : 2.1 (pvr 004e 1201)

timebase        : 512000000
platform        : PowerNV
model           : 8335-GTG
machine         : PowerNV 8335-GTG
firmware        : OPAL
MMU             : Radix
```

```
$ module use ${SOFTWARE_ROOT}/rhel/7.5/ppc64le/POWER9/modules/all
```

Different architectures

What about toolchain?

- XL C/C++
- XLF
- XL SMP
- MASS
- ESSL
- Platform MPI

Software

What we install with EasyBuild

- netCDF
- OpenFOAM
- WRF
- Bio stuff
- ...

What we do NOT install with EasyBuild

- ABAQUS
- MADYMO
- Pamcrash
- ...

www.hpcnow.com