TERAFLUX

Exploiting dataflow parallelism in Teradevice Computing

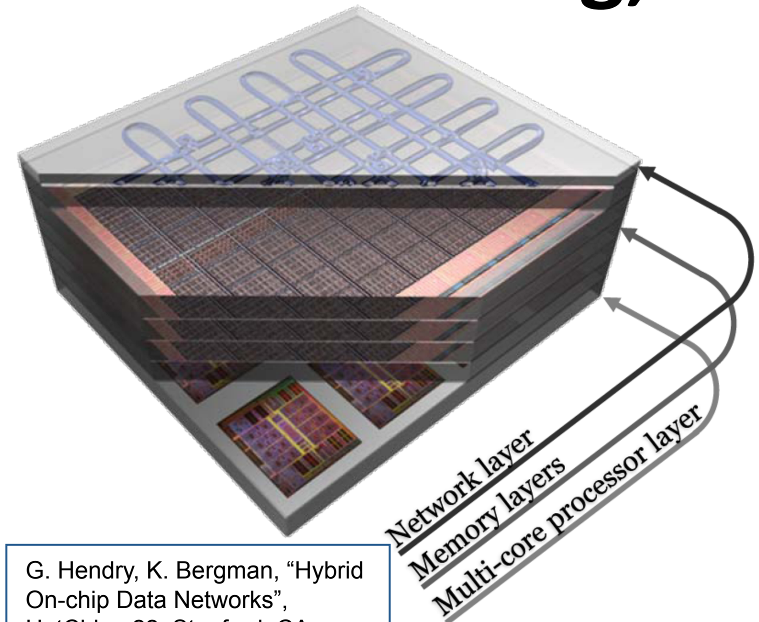
YEAR 2

What is **TERAFLUX** about

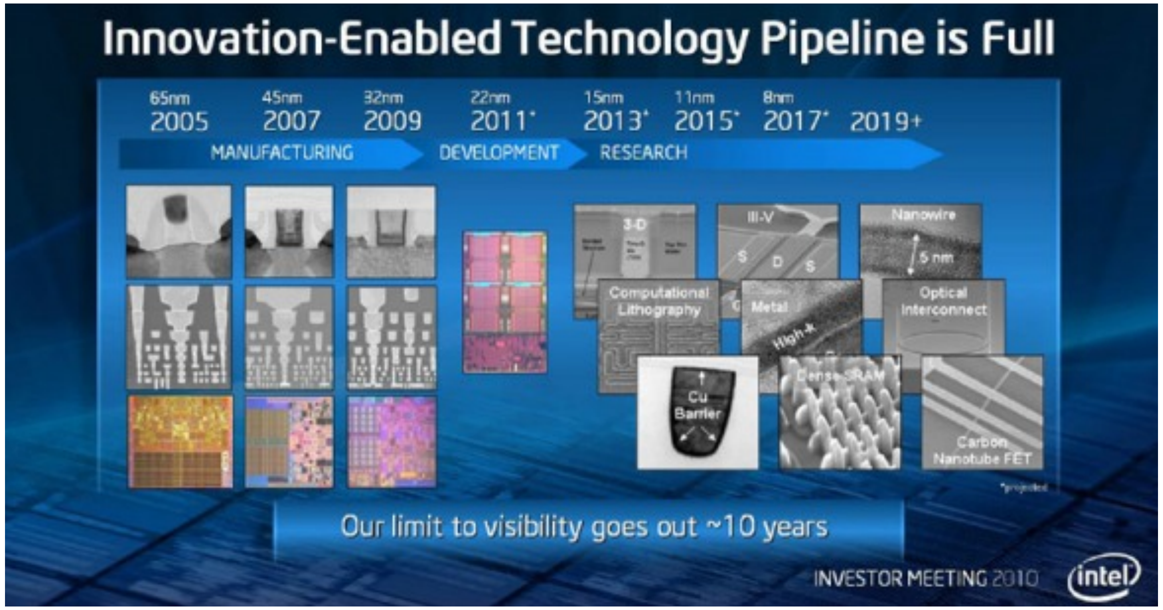
Architecture+Programmability+Reliability
of
Future (single chip)
Many-cores
(targeting 1000+ cores)

Future Scenarios

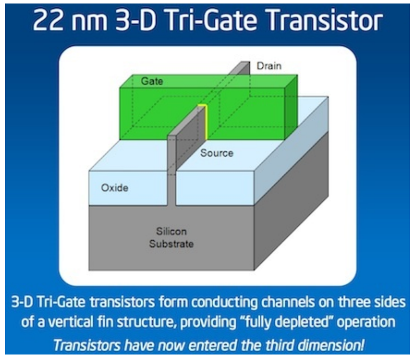
== 3D stacking, 8nm, 3D transistors, Graphene



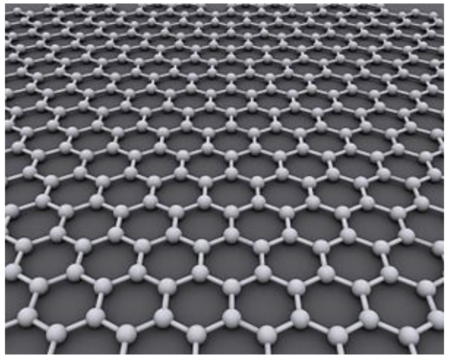
G. Hendry, K. Bergman, “Hybrid On-chip Data Networks”, HotChips-22, Stanford, CA – Aug. 2010

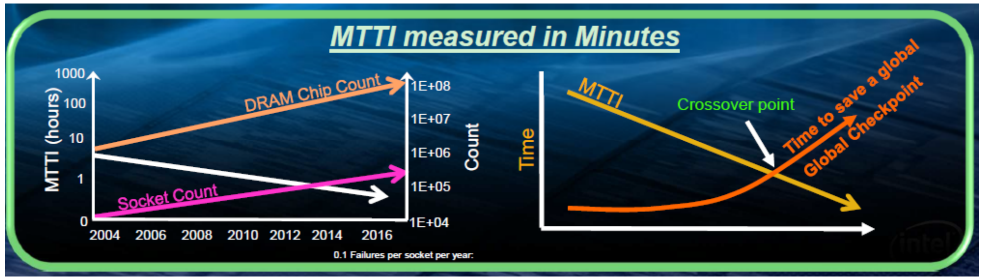


Fab D1X (OR), 42 (AZ) starting the 14nm node in 2013



22 nm 3-D Tri-Gate Transistor
Gate
Drain
Source
Oxide
Silicon Substrate
3-D Tri-Gate transistors form conducting channels on three sides of a vertical fin structure, providing “fully depleted” operation
Transistors have now entered the third dimension!





Pawloski, May 2011, Exascale Seminar, Ghent

TERA^F^LUX

Session Agenda

- *TERAFLUX: exploiting dataflow parallelism in teradevice computing - Year 2,* Roberto Giorgi (UNISI)
- *Teraflux Architecture,* Skevos Evripidou (UCY)
- *Reliability aspects in Teraflux,* Theo Ungerer (U. Augsburg)
- *Teraflux, from the programming model to the execution model* Antoniu Pop (INRIA)
- *Panel of questions about Teraflux Project*

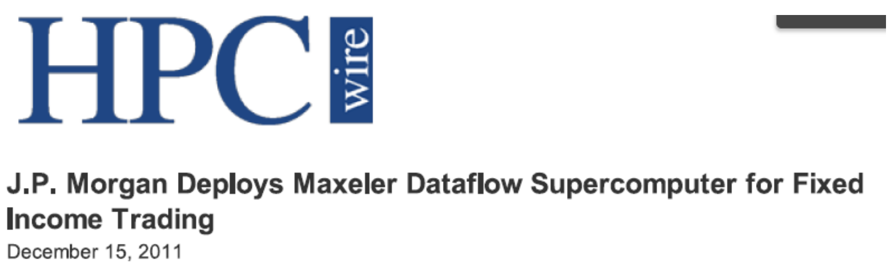
DATAFLOW

A Scheme of Computation in which an activity is initiated by presence of the data it needs to perform its function

(Jack Dennis)

Recent Projects/Efforts towards DATAFLOW

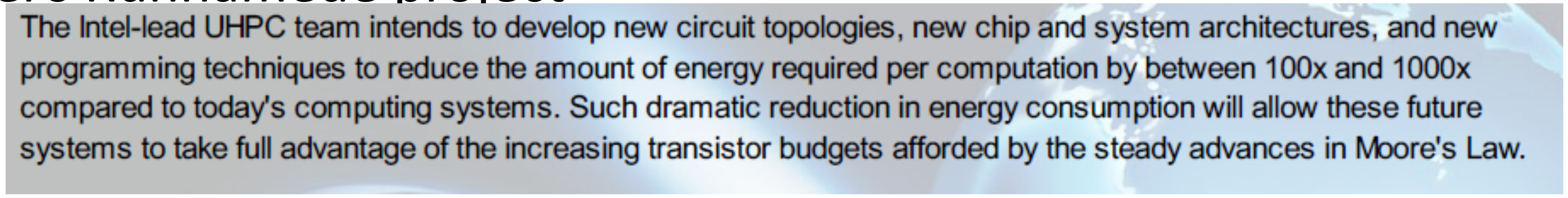
- Maxeler (UK) selling “dataflow computer” to J.P. Morgan → about 350x speedup vs. standard x86 cores


HPC wire

J.P. Morgan Deploys Maxeler Dataflow Supercomputer for Fixed Income Trading
December 15, 2011

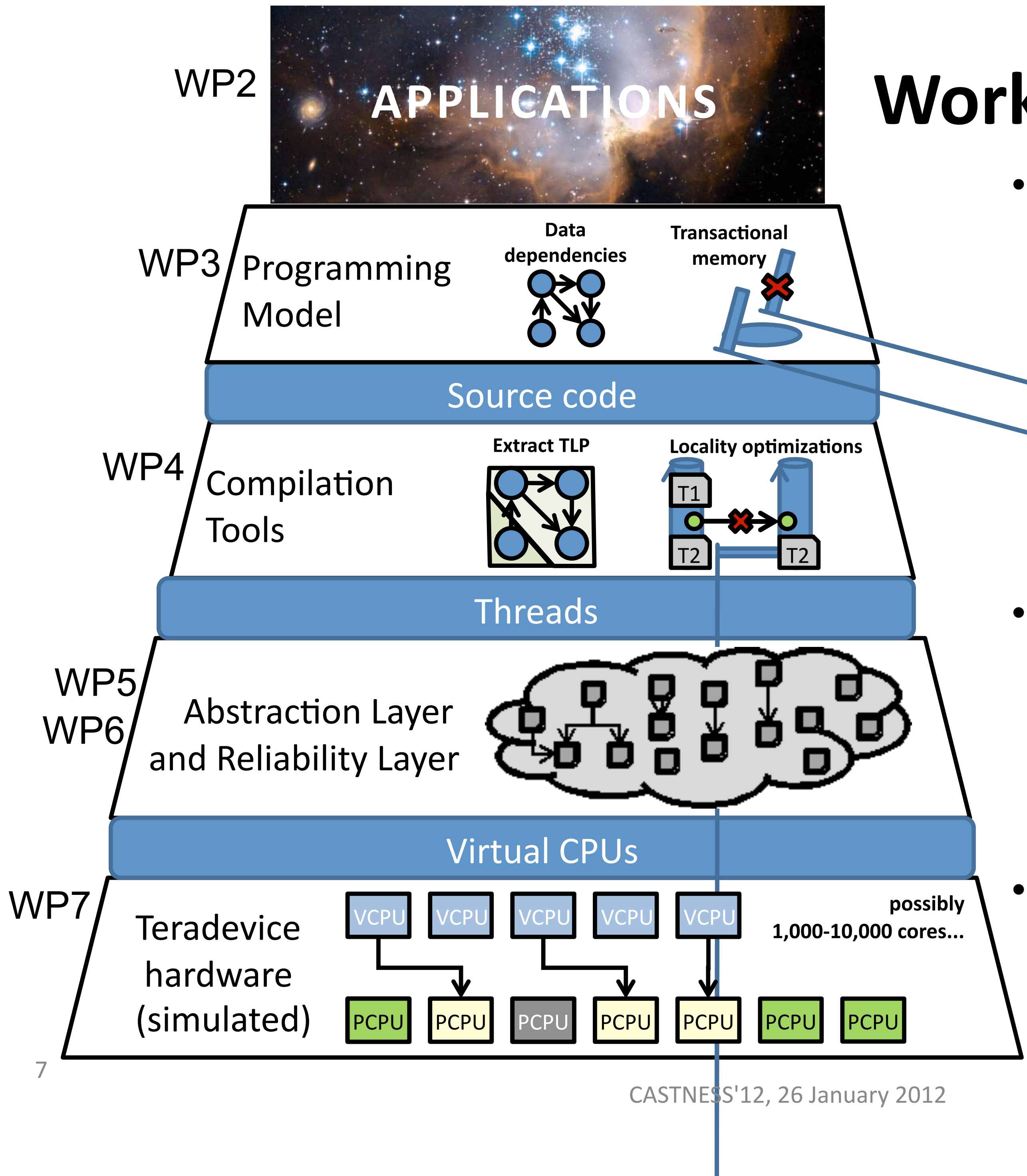
- DARPA funding 25M$ for UPHC program, encompassing:
  - Gao’s dataflow execution model (codelet based) – SWARM by ETI
  - Intel’s Runnamede project

UPHC=Ubiquitus
High-Performance Computing


The Intel-lead UHPC team intends to develop new circuit topologies, new chip and system architectures, and new programming techniques to reduce the amount of energy required per computation by between 100x and 1000x compared to today's computing systems. Such dramatic reduction in energy consumption will allow these future systems to take full advantage of the increasing transistor budgets afforded by the steady advances in Moore's Law.

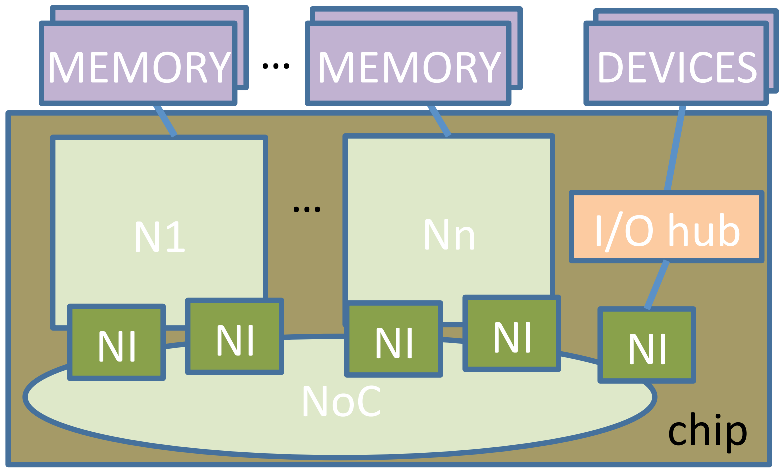
TERA^FLUX.EU

Working Hypothesis



- 1000 Billion- or 1 TERA-device computing platforms pose new challenges:
  - (at least) programmability, complexity of design, reliability
- TERAFLUX context:
  - High performance computing and applications (not necessarily embedded)
- TERAFLUX scope:
  - Exploiting a less exploited path (DATAFLOW) at each level of abstraction

TERAFLUX Architectural template



LEGENDA:

n = # of nodes

m = # of cores per node

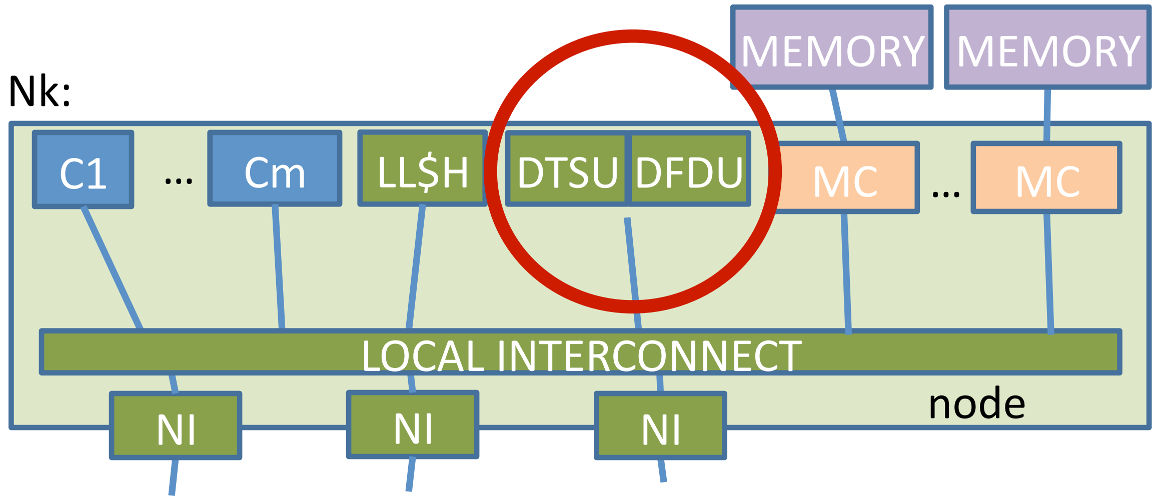
u= # of DRAM controllers insisting on the Unified Physical Address Space

z = # of I/O Hubs

Nk = k-th Node (k=1..n)

NI = Network Interface

NoC = Network on Chip



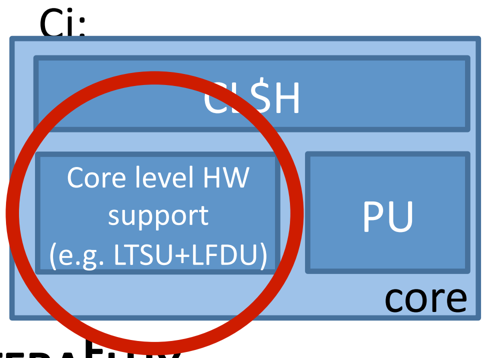
Cj = j-th core (j=1..m)

MC = Memory Controller

DTSU = Distributed Thread-Scheduler Unit

DFDU = Distributed Fault-Detection Unit

LL$H = Last Level Cache Hierarchy



CL$H = Core Level Cache Hierarchy

PU = Processing Unit

LTSU = Local Thread-Scheduler Unit

LFDU = Local Fault-Detection Unit

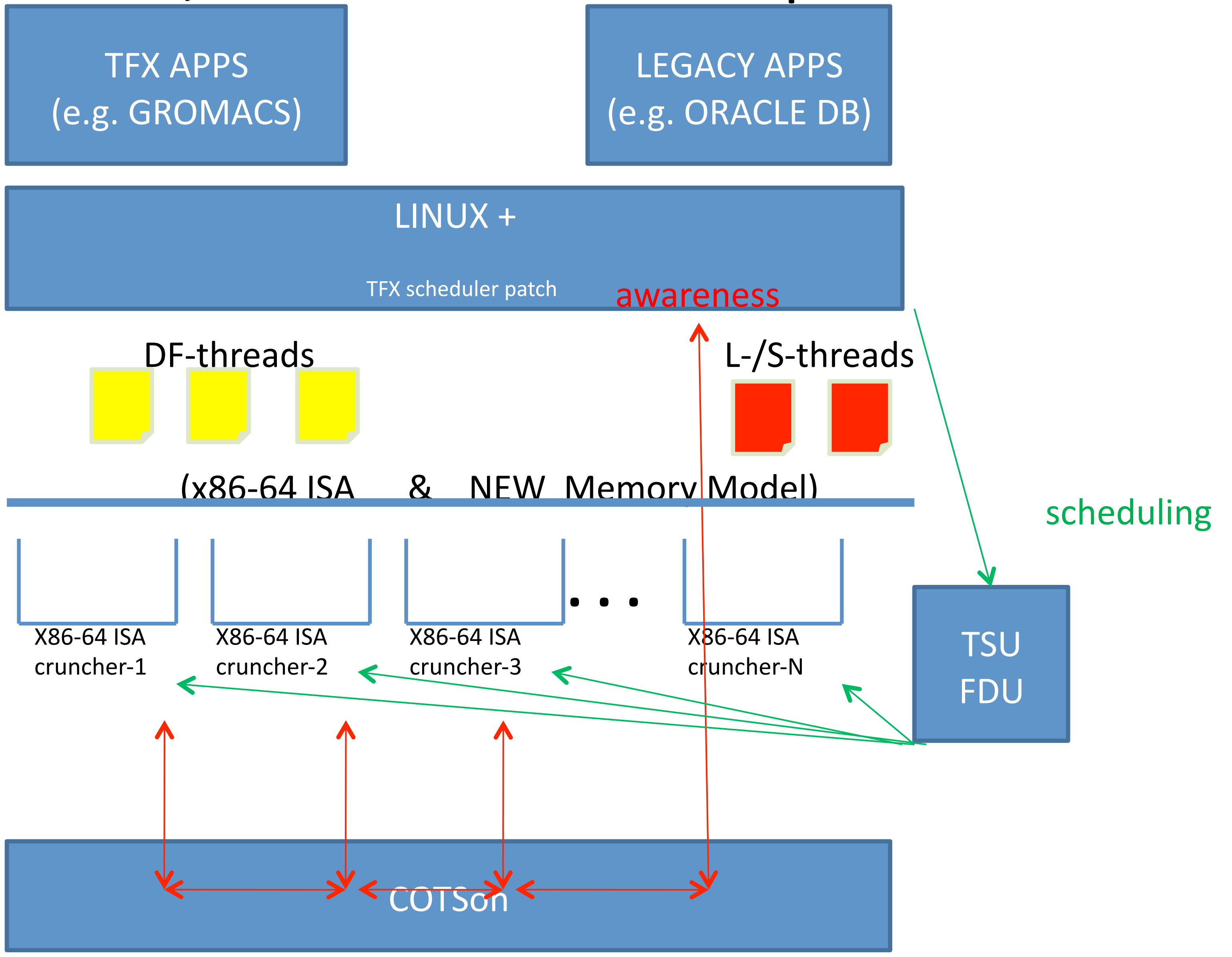
CHIP LEVEL	EXTERNAL or OTHER LAYER
NODE LEVEL	NODE OPTIONAL
CORE LEVEL	

TERAFLUX

Our pillars

- FIXED and MOST-USED ISA (**x86**)
- MANYCORE FULL SYSTEM SIMULATOR (**COTSon**)
- REAL WORLD APPLICATIONS (e.g. GROMACS)
- SYNCHRONIZATION: **TRANSACTIONAL MEMORY**
- **GCC** based TOOL-CHAIN
- OFF-THE-SHELF COMPONENTS FOR CORES, OS, NOC,MEMORY HIERARCHY
- **FDU** AND **TSU** (Fault Detection Unit and Thread Scheduling Unit)

Evaluating a MANY-CORE chip of the future (2020), i.e., 1000+ cores on a chip

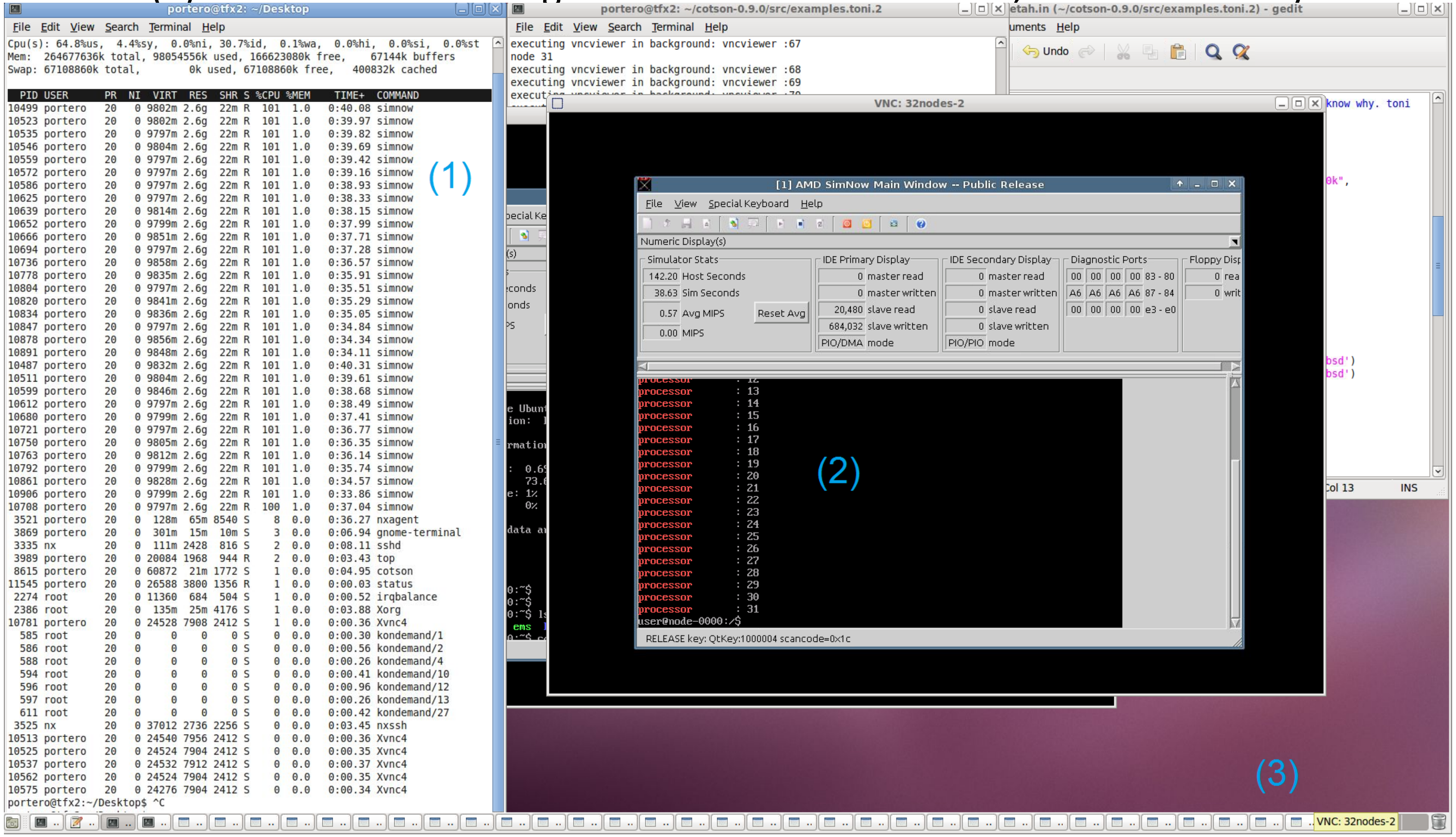


Simulator "illusion" (SW should only assume what seen at this level)

Simulator infrastructure

TERAFLUX

Simulation booting up 1024 cores. (1) COTSon execution of 32 SimNow instances. (2) Each instance manages 32 cores. Host: 48 cores, 256 GB memory


portero@tfx2: ~/Desktop
File Edit View Search Terminal Help
(1)
portero@tfx2: ~/cotson-0.9.0/src/examples.toni.2
etah.in (~/cotson-0.9.0/src/examples.toni.2) - gedit
VNC: 32nodes-2
[1] AMD SimNow Main Window -- Public Release
File View Special Keyboard Help
(2)
(3)

```
Cpu(s): 64.8%us,  4.4%sy,  0.0%ni, 30.7%id,  0.1%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:  264677636k total, 98054556k used, 166623080k free,    67144k buffers
Swap: 67108860k total,        0k used, 67108860k free,   400832k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
10499 portero   20   0 9802m 2.6g  22m R  101  1.0   0:40.08 simnow
10523 portero   20   0 9802m 2.6g  22m R  101  1.0   0:39.97 simnow
10535 portero   20   0 9797m 2.6g  22m R  101  1.0   0:39.82 simnow
10546 portero   20   0 9804m 2.6g  22m R  101  1.0   0:39.69 simnow
10559 portero   20   0 9797m 2.6g  22m R  101  1.0   0:39.42 simnow
10572 portero   20   0 9797m 2.6g  22m R  101  1.0   0:39.16 simnow
10586 portero   20   0 9797m 2.6g  22m R  101  1.0   0:38.93 simnow
10625 portero   20   0 9797m 2.6g  22m R  101  1.0   0:38.33 simnow
10639 portero   20   0 9814m 2.6g  22m R  101  1.0   0:38.15 simnow
10652 portero   20   0 9799m 2.6g  22m R  101  1.0   0:37.99 simnow
10666 portero   20   0 9851m 2.6g  22m R  101  1.0   0:37.71 simnow
10694 portero   20   0 9797m 2.6g  22m R  101  1.0   0:37.28 simnow
10736 portero   20   0 9858m 2.6g  22m R  101  1.0   0:36.57 simnow
10778 portero   20   0 9835m 2.6g  22m R  101  1.0   0:35.91 simnow
10804 portero   20   0 9797m 2.6g  22m R  101  1.0   0:35.51 simnow
10820 portero   20   0 9841m 2.6g  22m R  101  1.0   0:35.29 simnow
10834 portero   20   0 9836m 2.6g  22m R  101  1.0   0:35.05 simnow
10847 portero   20   0 9797m 2.6g  22m R  101  1.0   0:34.84 simnow
10878 portero   20   0 9856m 2.6g  22m R  101  1.0   0:34.34 simnow
10891 portero   20   0 9848m 2.6g  22m R  101  1.0   0:34.11 simnow
10487 portero   20   0 9832m 2.6g  22m R  101  1.0   0:40.31 simnow
10511 portero   20   0 9804m 2.6g  22m R  101  1.0   0:39.61 simnow
10599 portero   20   0 9846m 2.6g  22m R  101  1.0   0:38.68 simnow
10612 portero   20   0 9797m 2.6g  22m R  101  1.0   0:38.49 simnow
10680 portero   20   0 9799m 2.6g  22m R  101  1.0   0:37.41 simnow
10721 portero   20   0 9797m 2.6g  22m R  101  1.0   0:36.77 simnow
10750 portero   20   0 9805m 2.6g  22m R  101  1.0   0:36.35 simnow
10763 portero   20   0 9812m 2.6g  22m R  101  1.0   0:36.14 simnow
10792 portero   20   0 9799m 2.6g  22m R  101  1.0   0:35.74 simnow
10861 portero   20   0 9828m 2.6g  22m R  101  1.0   0:34.57 simnow
10906 portero   20   0 9799m 2.6g  22m R  101  1.0   0:33.86 simnow
10708 portero   20   0 9797m 2.6g  22m R  100  1.0   0:37.04 simnow
 3521 portero   20   0  128m  65m 8540 S    8  0.0   0:36.27 nxagent
 3869 portero   20   0  301m  15m  10m S    3  0.0   0:06.94 gnome-terminal
 3335 nx        20   0  111m 2428  816 S    2  0.0   0:08.11 sshd
 3989 portero   20   0 20084 1968  944 R    2  0.0   0:03.43 top
 8615 portero   20   0 60872  21m 1772 S    1  0.0   0:04.95 cotson
11545 portero   20   0 26588 3800 1356 R    1  0.0   0:00.03 status
 2274 root      20   0 11360  684  504 S    1  0.0   0:00.52 irqbalance
 2386 root      20   0  135m  25m 4176 S    1  0.0   0:03.88 Xorg
10781 portero   20   0 24528 7908 2412 S    1  0.0   0:00.36 Xvnc4
  585 root      20   0     0    0    0 S    0  0.0   0:00.30 kondemand/1
  586 root      20   0     0    0    0 S    0  0.0   0:00.56 kondemand/2
  588 root      20   0     0    0    0 S    0  0.0   0:00.26 kondemand/4
  594 root      20   0     0    0    0 S    0  0.0   0:00.41 kondemand/10
  596 root      20   0     0    0    0 S    0  0.0   0:00.96 kondemand/12
  597 root      20   0     0    0    0 S    0  0.0   0:00.26 kondemand/13
  611 root      20   0     0    0    0 S    0  0.0   0:00.42 kondemand/27
 3525 nx        20   0 37012 2736 2256 S    0  0.0   0:03.45 nxssh
10513 portero   20   0 24540 7956 2412 S    0  0.0   0:00.36 Xvnc4
10525 portero   20   0 24524 7904 2412 S    0  0.0   0:00.35 Xvnc4
10537 portero   20   0 24532 7912 2412 S    0  0.0   0:00.37 Xvnc4
10562 portero   20   0 24524 7904 2412 S    0  0.0   0:00.35 Xvnc4
10575 portero   20   0 24276 7904 2412 S    0  0.0   0:00.34 Xvnc4
portero@tfx2:~/Desktop$ ^C
```

```
executing vncviewer in background: vncviewer :67
node 31
executing vncviewer in background: vncviewer :68
executing vncviewer in background: vncviewer :69
```

Simulator Stats		IDE Primary Display		IDE Secondary Display	
142.20	Host Seconds	0	master read	0	master read
38.63	Sim Seconds	0	master written	0	master written
0.57	Avg MIPS	20,480	slave read	0	slave read
0.00	MIPS	684,032	slave written	0	slave written
		PIO/DMA	mode	PIO/PIO	mode

Diagnostic Ports				
00	00	00	00	83 - 80
A6	A6	A6	A6	87 - 84
00	00	00	00	e3 - e0

```
processor       : 13
processor       : 14
processor       : 15
processor       : 16
processor       : 17
processor       : 18
processor       : 19
processor       : 20
processor       : 21
processor       : 22
processor       : 23
processor       : 24
processor       : 25
processor       : 26
processor       : 27
processor       : 28
processor       : 29
processor       : 30
processor       : 31
user@node-0000:/$
```

RELEASE key: QtKey:1000004 scancode=0x1c

TERAFLUX

Note: the simulation is PARALLEL at GUEST NODE-LEVEL and it's also possible to distribute the simulation on several HOST NODES

Major Technical Innovations in TERAFLUX

- Fragmenting the Applications in Finer grained DF-threads:
  - DF-threads allow an easy way to decouple memory accesses, therefore hiding memory latencies, balancing the load, managing fault, temperature information without fine grain intervention of the software.
- Possibility to repeat the execution of a DF-thread in case this thread happened to be on a core later discovered as faulty
- Taking advantage of a “direct” dataflow communication of the data (through what we call DF-frames).
- Synchronizing threads while taking advantage of native dataflow mechanism (e.g. several threads can be synchronized at a barrier)
  - DF-threads allow (atomic) Transactional semantics (DF meets TM)
- A Thread Scheduling Unit would allow fast thread switching and scheduling, besides the OS scheduler; scalable and distributed
- A Fault Detection Unit works in conjunction with TSU

TERAFLUX SIMULATOR (COTSon)

http://cotson.sf.net

HP-Labs COTSon is **OPEN-SOURCE**

FUTURE AND EMERGING TECHNOLOGIES
PROJECT N. 249013

COOPERATION

SEVENTH FRAMEWORK PROGRAMME THEME
FET proactive 1 (ICT-2009.8.1)
Concurrent Tera-Device Computing

TERA^F^LUX

Exploiting dataflow parallelism in Teradevice Computing

PROJECT NUMBER: 249013

http://teraflux.eu