Exploring the use of light threads to improve the instruction level parallelism

D. González Márquez\textsuperscript{1}  A. Cristal Kestelman\textsuperscript{2}  E. Mocskos\textsuperscript{1}

\textsuperscript{1}Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Buenos Aires (C1428EGA), Argentina.

\textsuperscript{2}Barcelona Supercomputing Center, Artificial Intelligence Research Institute - CSIC, Barcelona (08034), Spain.

Mendoza, 30/7/2013
The research in processor architectures centers in optimizing the processor design based on performance, consumption, production cost, surface etc.
The research in **processor architectures** centers in optimizing the processor design based on *performance, consumption, production cost, surface* etc.

Parallelism allows the hardware to **accelerate applications** by executing *multiple, independent operations concurrently*. 
The research in processor architectures centers in optimizing the processor design based on performance, consumption, production cost, surface etc.

Parallelism allows the hardware to accelerate applications by executing multiple, independent operations concurrently.

Three levels:
- instruction-level parallelism (ILP)
- thread-level parallelism (TLP)
- data-level parallelism (DLP)
More cores are coming

The improvements in processor technology give us more available cores.

More cores are coming

The improvements in processor technology give us more available cores

But...

The applications and the programming models are not prepare for hundreds of cores.

We propose a change in the whole system

Two types of cores: full controlling cores and simple cores.
Simple example of interaction

mov $1, data_1
mov $2, data_2
mov $30, current_stack
mov new_stack, $30
mth_run
beq $0, jmp
mth_wait
jmp:
mov current_stack, $30
Core 0

Core mth

beq $0, jmp
mov $1, data_1
mov $2, data_2
mov $30, current_stack
mov new_stack, $30
mth_run
beq $0, jmp
mth_end
mth_wait
jmp:
mov current_stack, $30

E. Mocskos (DC-UBA)
Exploring the use of light threads
Mendoza, 30/7/2013
Testing the idea - Sorting an array using merge-sort

The new processor is completely simulated with gem5 based on ALPHA architecture. Instructions added: mth_run, mth_delegate, mth_end and mth_syn.
The unbalanced load in the four cores case comes from the implemented algorithm: two cores remain idle while the other two compute the last stage of the merge.
But we can deal with smaller problems

![Graph showing instructions count and clock ticks for Single Core, Dual Core, and Quad Core with different element counts: 64 elements and 32 elements.](image-url)
Conclusions and future work

- The proposal can effectively deal with very small problems.
- Standard tools need larger problems.
- Can execute small pieces of code in different simple processors.
Conclusions and future work

- The proposal can effectively deal with very small problems.
- Standard tools need larger problems.
- Can execute small pieces of code in different simple processors.
- New **processor organization** and **programming model**.
- Need to further test with **more processors**.
- Need to add support from **compilers** and **operating system**.
Questions?