An experience of early initiation to parallelism in the Computing Engineering Degree at the University of Murcia, Spain

Manuel E. Acacio, Javier Cuenca, Lorenzo Fernández, Ricardo Fernández-Pascual
Departamento de Ingeniería y Tecnología de Computadores
Joaquín Cervera, Domingo Giménez
Departamento de Informática y Sistemas
M. Carmen Garrido, Juan A. Sánchez Laguna
Departamento de Ingeniería de la Información y las Comunicaciones
José Guillén, Juan Alejandro Palomino Benito, María-Eugenia Requena
Centro de Supercomputación, Fundación Parque Científico, Murcia

EduPar Workshop, Shanghai, May 21, 2012

Outline

Parallel Computing today

- Computational systems are parallel: laptops, desktops, clusters, supercomputers, GPUs...
- But parallel computing is not sufficiently included in Computing Science studies, at least in Spanish universities, and in particular at the University of Murcia
- This Early Adopters project (Fall 2011) aims at improving this situation in the Computing Science Degree at the University of Murcia

Parallel Computing today

- Computational systems are parallel: laptops, desktops, clusters, supercomputers, GPUs...
- But parallel computing is not sufficiently included in Computing Science studies, at least in Spanish universities, and in particular at the University of Murcia
- This Early Adopters project (Fall 2011) aims at improving this situation in the Computing Science Degree at the University of Murcia

Parallel Computing today

- Computational systems are parallel: laptops, desktops, clusters, supercomputers, GPUs...
- But parallel computing is not sufficiently included in Computing Science studies, at least in Spanish universities, and in particular at the University of Murcia
- This Early Adopters project (Fall 2011) aims at improving this situation in the Computing Science Degree at the University of Murcia

University of Murcia

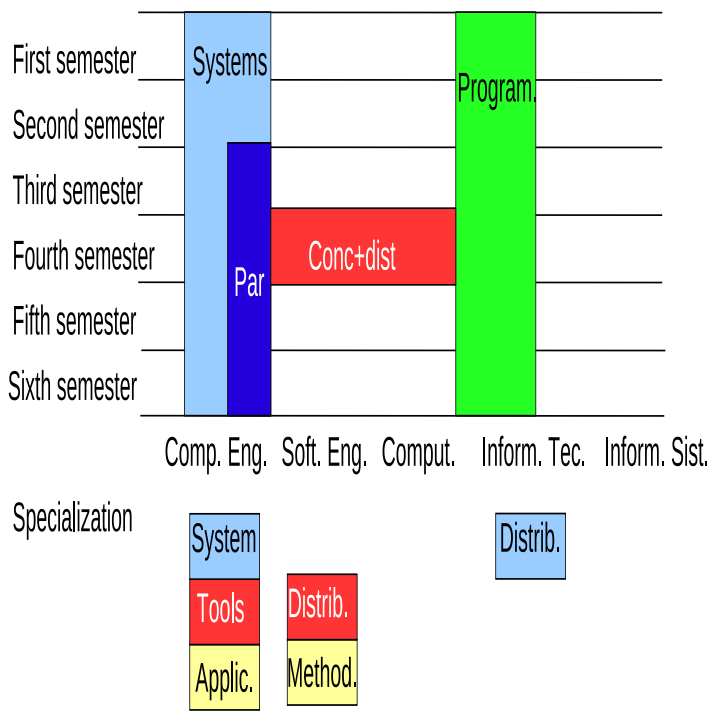
- Generalist University
- Approximately 31000 students

Computer Science school

- Computing Science Degree
  appr. 800 students + 80 teachers
- Master and PhD
  appr. 60 students
- 3 computing departments

Parallelism at the University of Murcia - present situation

- Parallelism in System courses from the third semester.
- Basic concepts of concurrency and distributed computing in a programming course in the fourth semester.
- Algorithmic aspects are not studied in any compulsory course.
- Intensification in parallelism in some specializations, but parallelism is not included in all the specializations.



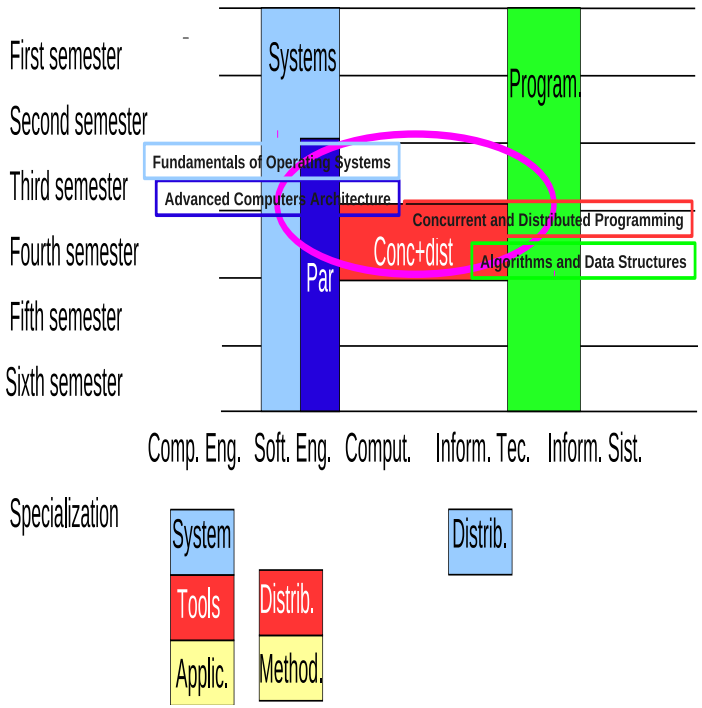
$\Rightarrow$ Computer Science students at the University of Murcia can obtain their degree without having developed and optimized any parallel code.

Parallelism at the University of Murcia - the project

- Four compulsory courses in the second year of the degree.

  Two courses already included parallelism (ACA and CDP).

  Two courses include topics of parallelism for the first time (FOS and ADS).
- Systems (ACA and FOS) and Programming (CDP and ADS) courses, with topics in the four parallelism aspects of the IEEE TCPP curriculum.
- Three departments and a Computing Centre: coordinated treatment of the topics + use of a range of computational systems in the practicals.



Courses involved

- Fundamentals of Operating Systems
  Processes, Memory, Files, I/O, Security, Shell Scripts, Users Management, File systems, Backups, Monitoring
- Advanced Computer Architecture
  Performance Analysis, Pipelining, Control Dependencies, Static and Dynamic Scheduling of Instructions, Memory System Organisation and Performance
- Algorithms and Data Structures
  Analysis of Algorithms, Complexity, Greedy Algorithms, Backtracking, Branch & Bound, Game Trees, Divide and Conquer, Dynamic Programming
- Concurrent and Distributed Programming
  Loosely and Strongly Coupled Systems Programming, Classic Programming Paradigms in Distributed Systems

Courses involved

- Fundamentals of Operating Systems
  Processes, Memory, Files, I/O, Security, Shell Scripts, Users Management, File systems, Backups, Monitoring
- Advanced Computer Architecture
  Performance Analysis, Pipelining, Control Dependencies, Static and Dynamic Scheduling of Instructions, Memory System Organisation and Performance
- Algorithms and Data Structures
  Analysis of Algorithms, Complexity, Greedy Algorithms, Backtracking, Branch & Bound, Game Trees, Divide and Conquer, Dynamic Programming
- Concurrent and Distributed Programming
  Loosely and Strongly Coupled Systems Programming, Classic Programming Paradigms in Distributed Systems

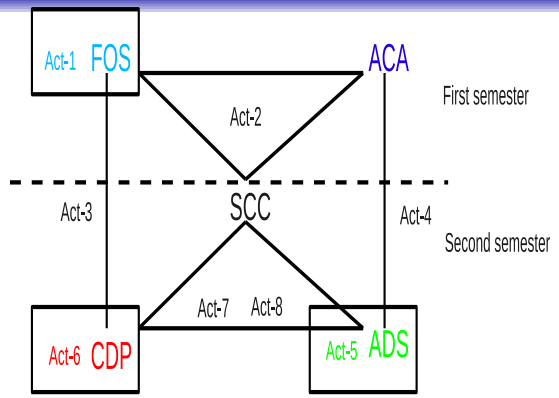
Courses involved

- Fundamentals of Operating Systems
  Processes, Memory, Files, I/O, Security, Shell Scripts, Users Management, File systems, Backups, Monitoring
- Advanced Computer Architecture
  Performance Analysis, Pipelining, Control Dependencies, Static and Dynamic Scheduling of Instructions, Memory System Organisation and Performance
- Algorithms and Data Structures
  Analysis of Algorithms, Complexity, Greedy Algorithms, Backtracking, Branch & Bound, Game Trees, Divide and Conquer, Dynamic Programming
- Concurrent and Distributed Programming
  Loosely and Strongly Coupled Systems Programming, Classic Programming Paradigms in Distributed Systems

Courses involved

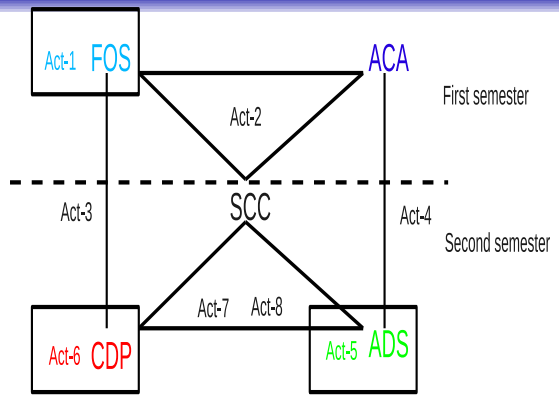
- Fundamentals of Operating Systems
  Processes, Memory, Files, I/O, Security, Shell Scripts, Users Management, File systems, Backups, Monitoring
- Advanced Computer Architecture
  Performance Analysis, Pipelining, Control Dependencies, Static and Dynamic Scheduling of Instructions, Memory System Organisation and Performance
- Algorithms and Data Structures
  Analysis of Algorithms, Complexity, Greedy Algorithms, Backtracking, Branch & Bound, Game Trees, Divide and Conquer, Dynamic Programming
- Concurrent and Distributed Programming
  Loosely and Strongly Coupled Systems Programming, Classic Programming Paradigms in Distributed Systems

Tasks



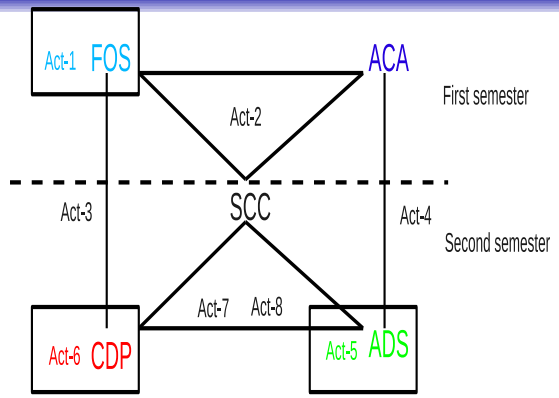
- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Tasks



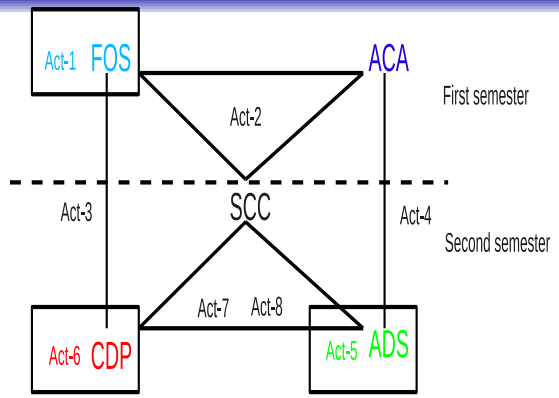
- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Tasks



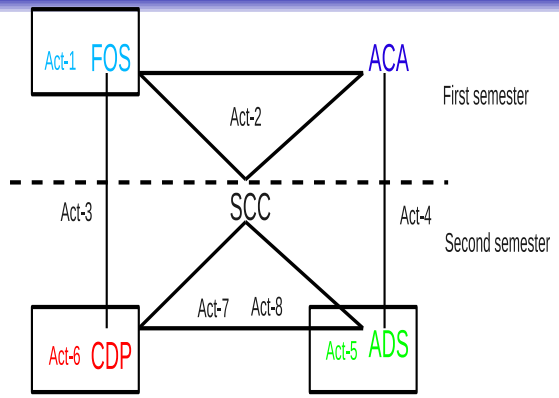
- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Tasks



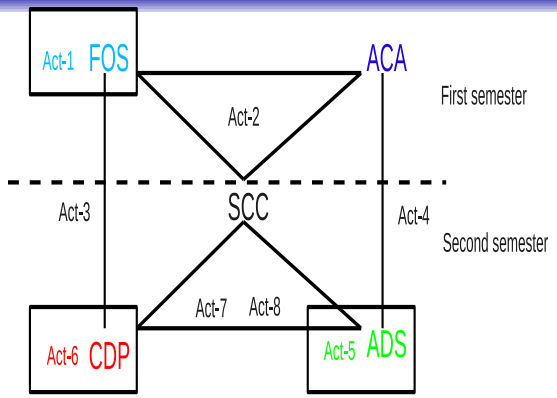
- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Tasks



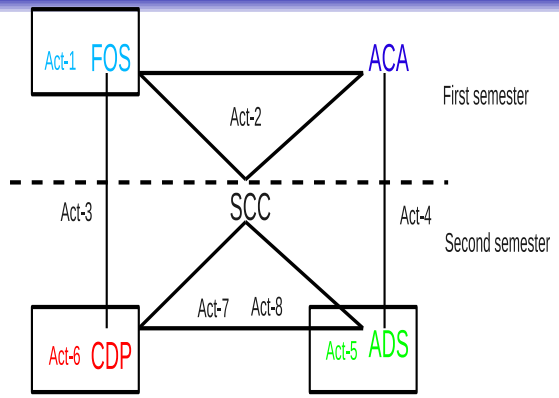
- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Tasks



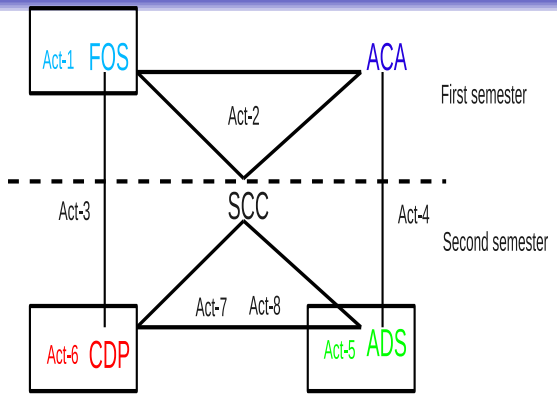
- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Tasks



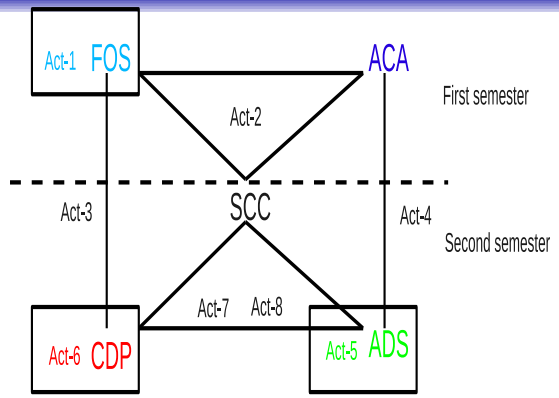
- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Tasks



- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Tasks



- **Act-1**, FOS: threads management and monitoring. (Practicals)
- **Act-2**, FOS+ACA+SCC: computing centre. (Visit)
- **Act-3**, FOS+CDP: performance management of threads and processes. (Practicals)
- **Act-4**, ACA+ADS: influence of memory hierarchy on performance. (Practicals)
- **Act-5**, ADS: parallel algorithmic schemes and cost of parallel algorithms. (Seminars)
- **Act-6**, CDP: basic shared-memory and message-passing constructors. (Theory and practicals)
- **Act-7**, CDP+ADS+SCC: shared-memory programming. (Practicals)
- **Act-8**, CDP+ADS+SCC: message-passing programming. (Practicals)

Architecture

Topic			Previous		First semester		Second semester					
			ACA	CDP	1	2	3	4	5	6	7	8
Architecture classes	0.5	C		X			X			X	X	X
Superscalar	0.5	C	X									
SIMD/Vector	0.5	K	X									
Pipelines	6	C	X									
OoO execution	4	C	X									
Multicore	1	K			X		X					
NUMA (shared memory)	0.5	K			X	X	X					
Cache organization	6	K	X									
Atomicity	2	C		X			X			X	X	X
Impact memory hier. on soft.	3.5	A	X		X			X				
Cycles per instr. Benchmarks	0.5	C	X									
Benchmarks	0.5	C	X									
Spec marks	0.5	C	X									
Peak performance	0.5	K	X									
MIPS/FLOPS	1	C	X					X	X		X	X
Sustained performance	0.5	K	X									

- Most of the architecture topics were studied in ACA.
- A few topics are included, and others are treated more in depth and in collaboration with different courses.

Programming

Topic			Previous		First semester		Second semester					
			ACA	CDP	1	2	3	4	5	6	7	8
Shared memory	12	A								X	X	
Distributed memory	12	C								X		X
Client server	0.5	C		X							X	X
Task/thread spawning	2.5	A		X							X	X
SPMD	2	C		X							X	X
Shared memory notations	10	A		X						X	X	
Language extensions	1	K		X								
Libraries	10	A		X			X					
SPMD notations	3	A		X						X	X	
MPI	3	C								X		X
Semantic tasks and threads	5.5	C		X	X						X	X
Synchronization	2	A		X			X			X	X	X
Critical regions	2.5	A		X					X	X	X	
Producer-consumer	1.5	A		X						X	X	X
Monitors	4	A		X								
Deadlocks	0.5	K		X								
Memory models	0.5	K		X								
Scheduling and comp.	2	C	X									
Decomposition strategies	1	K	X									
Loop fusion	0.5	A	X									
Scheduling and mapping	3	C	X		X							
Performance monitoring	2	A			X							
Performance metrics	1.5	C	X						X		X	X
Speed-up	2	C	X						X		X	X
Efficiency	1	C	X						X		X	X
Amdahl's law	1	C	X									

- Most of the programming topics were studied in CDP.
- Programming topics are put in practice.

Algorithms

Topic			Previous		First semester		Second semester					
			ACA	CDP	1	2	3	4	5	6	7	8
Asymptotics cost.	0.5	C							X			
Time	0.5	C							X			
Space	0.5	C							X			
Speed-up	0.5	C							X		X	X
Notions from scheduling	1	K		X					X		X	X
Divide and Conquer	0.5	A								X	X	X
Broadcast	0.5	K		X								
Asynchrony	1	K		X								X
Synchronization	1	A		X								
Sorting	0.5	A							X		X	X
Graph search	0.5	K							X		X	X
Specialized computations	1	K						X	X		X	X

- Most algorithmic topics are new.
- They are studied in CDP and joint practicals are done with ADS.
- Students develop and theoretically and experimentally analyse simple parallel programs in multicore and clusters.

Cross Cutting

Topic			Previous		First semester		Second semester					
			ACA	CDP	1	2	3	4	5	6	7	8
Why and what is PDC.	1	C		X								
Concurrency	1	C		X								
Non-determinism	1	A		X								
Power	0.5	K	X									
Locality	1.5	C	X		X							
Security in Dist. systems	1	K			X	X						

- Wider vision of different aspects of parallelism.
- Collaboration of the Supercomputing Centre, with a visit and presentation of the laboratory: security, management, configuration, applications...

Courses - hours

Approximate number of hours devoted in each course to each part of the curriculum

Previously:

	Arch.	Prog.	Algor.	Cross Cut.	TOTAL	Percentage
FOS	0	0	0	0	0	0
ACA	21	13	0	2	36	60
CDP	1	42	2.5	3	48.5	78
ADS	0	0	0	0	0	0
TOTAL	22	55	2.5	5	84.5	12

With the project:

	Arch.	Prog.	Algor.	Cross Cut.	TOTAL	Percentage
FOS	1	5	0	0.5	6.5	12
ACA	24	13	0	2.5	37.5	62
CDP	1.5	55	2.5	3	62	100
ADS	1.5	13.5	5.5	0	20.5	32
TOTAL	26	86.5	8	6	126.5	19

Evaluation

	Students	Project
Act-1	laboratory + test	opinion and test satisfactory
Act-2	not assessed	high participation
Act-3	theory + laboratory	high participation
Act-4	laboratory + practicals	ongoing, high participation
Act-5	practicals	ongoing, high participation
Act-6	theory + laboratory	high participation
Act-7	laboratory + practicals	ongoing, students of the non participating group ask to participate
Act-8	laboratory + practicals	ongoing, students of the non participating group ask to participate

Subjective appreciation

- Positive experience, with participation of the students in non compulsory activities.
- 2 of the 3 groups participate, for next year it could be interesting to extend the experience to all the groups.
- Difficulties with rigid Degree plan and teachers not familiar with parallel computing.
- But some teachers without previous experience in parallelism have joined the experience.
- Most activities non compulsory, so attendance at the activities is satisfactory, but the active participation (homework, practicals...) is low.

Subjective appreciation

- Positive experience, with participation of the students in non compulsory activities.
- 2 of the 3 groups participate, for next year it could be interesting to extend the experience to all the groups.
- Difficulties with rigid Degree plan and teachers not familiar with parallel computing.
- But some teachers without previous experience in parallelism have joined the experience.
- Most activities non compulsory, so attendance at the activities is satisfactory, but the active participation (homework, practicals...) is low.

Subjective appreciation

- Positive experience, with participation of the students in non compulsory activities.
- 2 of the 3 groups participate, for next year it could be interesting to extend the experience to all the groups.
- Difficulties with rigid Degree plan and teachers not familiar with parallel computing.
- But some teachers without previous experience in parallelism have joined the experience.
- Most activities non compulsory, so attendance at the activities is satisfactory, but the active participation (homework, practicals...) is low.

Subjective appreciation

- Positive experience, with participation of the students in non compulsory activities.
- 2 of the 3 groups participate, for next year it could be interesting to extend the experience to all the groups.
- Difficulties with rigid Degree plan and teachers not familiar with parallel computing.
- But some teachers without previous experience in parallelism have joined the experience.
- Most activities non compulsory, so attendance at the activities is satisfactory, but the active participation (homework, practicals...) is low.

Subjective appreciation

- Positive experience, with participation of the students in non compulsory activities.
- 2 of the 3 groups participate, for next year it could be interesting to extend the experience to all the groups.
- Difficulties with rigid Degree plan and teachers not familiar with parallel computing.
- But some teachers without previous experience in parallelism have joined the experience.
- Most activities non compulsory, so attendance at the activities is satisfactory, but the active participation (homework, practicals...) is low.

Additional information

- A poster in the posters' session: 16:15-18:00
- The paper in the proceedings describes in more detail how each topic is treated.
- Project website: `http://www.um.es/earlyadopters`
- ... or my e-mail `domingo@um.es`
- ... or **questions here**