

SPD 2011 –12 Course Introduction

Strumenti di programmazione per sistemi paralleli e distribuiti (SPD)

~

Programming Tools for Distributed and Parallel Systems

M. Coppola massimo.coppola@isti.cnr.it

- Programming Tools for Parallel and Distributed Systems (SPD)
 - 1st term (Sept. 2011- Dec. 2012)
 - 9 credits
 - 72 hours : ~48 lessons, ~24 laboratory
 - Final test: lab project + oral examination
 - Includes discussing the project

– Course wiki :

[www.cli.di.unipi.it/doku/magistraleinformaticanetworking/
spd](http://www.cli.di.unipi.it/doku/magistraleinformaticanetworking/spd)

Description and Analysis of parallel and distributed programming platforms and models, to tackle problems of daunting size, scale and performance requirements

Parallelism at different levels of scale

- *Theoretical foundations*
- Standards for platforms and programming systems
- State-of-the-art solutions
- Practical use
- *Applications*

Course topics

- Parallel programming tools & platforms for HPC
 - HPC and also large scalable systems: Grids, Clouds
- Many different parallelism levels
 - Many-core systems
 - Multiprocessor systems
 - Clusters
 - Clouds & Grids

- MPI – Message Passing Interface
 - message passing standard
 - Cluster and Cloud computing
 - linked library
 - Support for several languages
 - C, C++, Fortran + several more from 3rd parties
- TBB – Intel-Thread Building Blocks library
 - C++ template library
 - shared memory
 - multiple threads
 - aims at multi-core CPUs

- ASSIST
 - High-Level SPP language for Clusters/Clouds
 - dynamic and autonomic management
- GPGPU
 - General Purpose GPU programming
 - Exploit Many-core on-chip parallelism targeted at graphics for general purpose programs
 - High-level approaches like CUDA and Brooks+
 - APU development: soon to merge with standard programming?

- Clouds, Clusters, multi / many-core systems
 - XtremOS
 - Grid-enabled O.S.
 - Based on Linux
 - Provides an “abstract machine” view over a geographic network
 - Contrail
 - Federation of Clouds
- OpenNebula
 - A specific Cloud Platform
 - Open Source European Platform
 - Implementation and APIs

Links to other courses

- **SPA** is a prerequisite
 - High-performance Computing Systems and Enabling Platforms
- **SPM** Distributed systems: paradigms and models
 - SPM theoretical foundations, surveys of systems
 - SPD focuses on few programming systems + lab time
- **CPA** Complements of Distributed Enabling Platforms
 - CPA focuses on Cloud/Grid platforms, related programming tools
- **LPD** Laboratory of Distributed Software Engineering
 - Tools and methodology for distributed software design and development
- **ALP** Parallel & Distributed Algorithms
 - ALP provides basics of parallel algorithmic cost models

QoS an SLA in {networking, virtualization, services}

- **SRT** Real time systems
 - Real-time theory has applications to SLA
- **P2P** Peer to Peer Systems
- **Network Optimization Methods**
 - QoS Routing and Scheduling

- On SPD topics
 - Research oriented [R], implementation [I], or both [IR]

QoS control in Federated Clouds: Service Level Agreement, Quality of Protection, Hypervisor enhancement for QoS in the Contrail project	R
Algorithms and heuristics resource management in Cloud Federations in the Contrail project	R I
Software Virtual Machines integration in Cloud platforms	IR
Distributed JIT compilation for multicore CPUs	IR
Automatic Data Extraction and Analysis from distributed Medical Administrative Databases	R I

- Other joint proposals may pop out in collaboration with P2P (Prof. Laura Ricci) and other courses related to SPD

Timetable

- 6 hours per week (standard)
 - Starting on 19/9/2011
 - Some weeks will be skipped due to work constraints in Sept. – Oct.
 - 35 lessons or less
- We will need to set up at least 4 extra lessons, possibly in October.
- 8 hours per week in October
→ still ~ 24 hours per month.

- No lab facilities explicitly set up yet
 - Usually each student has his own laptop
- Ok for development with most of the programming tools (MPI, TBB, GPGPU, etc...)
- For testing, options are
 - Labs of the C.S. Dept. used as a cluster
 - Labs in scuola S.Anna
 - Other machines on a case-by-case basis

- Coding individual project
 - Agree topic with the teacher, write 2-page summary
 - Project will use at least one of the frameworks and tools presented
 - E.g. MPI, or TBB+MPI, or OpenCL + TBB ...
 - Submit project and a written report on it
 - explaining problem, approach, test results
 - Discuss project
 - may be in seminar form with the class, if so agreed
- Oral test
 - About any topic in the course program