

Advanced programming models for distributed/grid computing

Marco Danelutto

Dept. Computer Science - Univ. of Pisa

“Advanced programming models” topic leader - ERCIM CoreGRID WG

Main research challenges

- **Advanced programming models for Services**
 - *providing higher level abstractions for service programming in distributed architectures*
 - *closer to the programmer than to the machine*
 - *empowering application programmers to rapid prototype and tune possibly complex service programs*
- **Autonomic Management of Service Applications**
 - *support of autonomic management of self-* properties in the advanced programming models*
 - *suitable, configurable abstractions provided to the final users*
 - *co-design of HL programming abstractions & autonomic management*

... in more detail

- **Parametric programming patterns (reusable, efficient)**
 - *design patterns, skeletons, coordination languages, (service) orchestration, choreography, ...*
- **High level, general purpose, unconventional programming models**
 - *bio-inspired, non-deterministic, non control flow, ...*
- **High level patterns for autonomic management of distributed applications**
 - *rule based autonomic management, self-* heuristics, ...*
- **Coordinated (hierarchical), multi concern autonomic management of distributed applications**
 - *distributed agreement, distributed coordination, self-* heuristics, ...*

Participants (14 groups in 12 institutions)



University of Pisa



Rennes, Rhône Alpes,
Sophia Antipolis



Science and Technology
Facilities Council



ISTITUTO DI SCIENZA E TECNOLOGIE
DELL'INFORMAZIONE "A. FAEDO"

ISTI/CNR Pisa



Haute Ecole Spécialisée de
Suisse Occidentale



Muenster University



Queen's Univ.
Belfast



Innsbruck University



University of
Westminster



Poznan Supercomputing and
Networking Center



High Performance Computing
Center Stuttgart

Sample topic: autonomic managers

- **Autonomic management of performance tuning in GCM structured parallel components (BS)**
 - *single non functional concern management through rule based autonomic managers*
 - *hierarchical management of single non functional concern*
 - *methodology to handle multiple non functional concerns*
 - *distributed coordination of manager decisions*
 - *generalization to generic concern management*
- **Different non functional concerns and SLA management**



Sample topic:

multi-level parallel&distributed programming

- **Parallel/distributed programming models exposing different levels of the parallel target architecture through proper high level paradigms**
 - *GPU becoming more and more popular, ubiquitous and efficient (nVIDIA → Larrabee → ...)*
 - *Synergistic processing units (Cell and the alike)*
 - *Dynamic (multi) core pairing through hyper channels*
- **co processor programming still at the assembly level**
 - *possibility to exploit libraries (à la TBB)*
 - *possibility to exploit structured programming implementation techniques*



Sample topic: MapReduce in-the-large

- **Middleware support for MapReduce techniques originally developed in the skeleton community, recently moved to distributed computing arena by Google**
 - *still to be properly exploited in the Grid scenario*
 - *opens new perspectives relative to the support of structured programming patterns*
 - *possible integration in the GCM BS framework*
 - *exploiting the autonomic management facilities*



Sample topic: abstractions for services

- **Patterns (skeletons) and operators for distributed component/service oriented computing**
- **Group abstractions for managing distribution, dynamism and scale**
 - *orthogonal perspective on service computing patterns*
 - *structuring service operations through groups*
 - *collective operations within a group*
 - *structured interaction among groups*
 - *autonomic management of group activities*



Sample topic: innovative computing models

- **Chemical computing**

- *rule based structured computations*
- *modelling both bio-* & more traditional computations*
- *explored for*
 - efficiency
 - portability
 - expressive power/programmability



Science and Technology
Facilities Council

Sample topic: real time & reliable services

- **Typical paradigms for real time, interactive services**
 - *increasingly important*
- **Fault-tolerant parallel services**
 - *fail-stop, byzantine and malicious contexts*
 - *provable multi-criteria performance for the deployment and execution on cloud computing platforms*
- **increasingly interesting topics**
 - *after initial pioneer service frameworks*
 - *when targeting particular markets (e.g. games, financial, ...)*



Topic Initiatives

- Work in progress workshop
 - *TBD*
(e.g. 5-6 or 26-27 Nov. in Pisa)

- research groups participating to the topic present more pertinent research activities / projects
- single researchers present work in progress related to “Advanced programming models for services”
- Electronic proceedings on the web site, possibly as an ERCIM/CoreGRID TR (TBD) to enforce dissemination

November 2009						
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
26	27	28	29	30	31	1
2	3	4	5	6	7	8
			ERCIM CoreGRID WG			
9	10	11	12	13	14	15
		• Facoltà				
16	17	18	19	20	21	22
23	24	25	26	27	28	29
			ERCIM CoreGRID WG			
30	1	2	3	4	5	6

Topic initiatives (2)

- Call 5 projects
 - *cooperation among topic partners*
 - *addressing several points in the call*

Objective ICT-2009.1.2: Internet of Services, Software and Virtualisation

...

b) **Highly Innovative Service / Software Engineering**

- *Service / Software engineering methods and tools* covering automatic support at run-time for decisions and changes that are currently adopted at design time. Focus is on innovative approaches to very large, dynamic open service networks, user development of services/software, systems evolvability and acquisition, reasoning and incorporation of domain knowledge in all phases of the service/software life cycle. High-level description and executable languages for services/software with support for adaptation and technologies for improving system response time, performance and throughput are in the scope of the research,

Objective ICT-2009.8.5: FET proactive 5: Self-Awareness in Autonomic Systems

The challenge is to create computing and communication systems that are able to optimise overall performance and resource usage in response to changing conditions, adapting to both context (such as user behaviour) and internal changes (such as topology). To achieve this, autonomic systems should enable nodes to build up an awareness relating to higher and even global levels, e.g. of patterns of use, system performance, network conditions and available resources. This requires breaking through the tradition of fixing abstraction layers at design time, which hide issues at lower layers (e.g., by hiding mobility, heterogeneity, or drops in performance), but inevitably limit the scope for optimising resource usage and responding to changing conditions.

Target Outcome

New concepts, architectures, foundations and technologies for:

- a) **Creating awareness** at the level of autonomic nodes, by allowing them to interactively and selectively collect information about the system, and use it effectively. A central question is how to link awareness of performance, conditions, available resources, etc., to the nature of information that is exchanged.
- b) **Dynamic self-expression**, namely the ability to autonomically use awareness to adapt the trade-off between abstraction and optimisation. There is a need for understanding the consequences of this principle on system behaviour and performance, and designing/experimenting with related features.

Contacts

- marcod@di.unipi.it
- list of addresses collected and used to disseminate information (up to now) → official mailing list after this workshop
- web page on CoreGRID web site
<http://www.coregrid.net/mambo/content/view/770/431/>
- web page at di.unipi.it (wiki)
<http://cotognata.di.unipi.it/~marcodanelutto/wiki/doku.php?id=advancedprogrammingmodelsercimcoregrid>

Topic WIKI / WEB

CoreGRID Network of Excellence – European Grid Research – Advanced programming models for Services

http://www.coregrid.net/mambo/content/view/770/431/

search...

European Research Network on Foundations, Software Infrastructure for large scale distributed, GRID and Peer-to-Peer Technologies

Home > CoreGRID WG > Research Topics > Advanced programming

The CoreGRID Network of Excellence currently offers

- **Fellowships:** for postgraduate students in the field of GRID Research
- **Job announcements:** related to GRID research free of charge

MAIN MENU

- Home
- News
- Events
- CoreGRID WG**
 - Chair & Advisory Board
 - Core members
 - Membership
 - Research Topics
 - Service Level Agreement
 - Data & Knowledge Mgt
 - Scheduling
 - Virtualization
 - Network monitoring
 - Volunteer Computing
 - Trust & Security
 - Self-* and adaptive mech.
 - Advanced programming
 - Tools and Environments
 - WG Meetings
 - CoreGRID ERCIM Working Group Workshop
- CoreGRID NoE**
- Institutes**
- Integration Activities**
- Dissemination**
- Training & Education**
- CoreGRID & Industry**
- Mobility Portal**
- Trust&Security Portal**
- Collaboration Gateway**
- Other Collaborations**
- Links**
- Contact Us**

marcod's home page/~marcodanelutto/wiki/doku.php?id=advancedprogrammingmodelsercimcoregrid

http://cotognata.di.unipi.it/~marcodanelutto/wiki/doku.php?id=advancedprogrammingmodelsercimcoregrid

Advanced programming models for Services

A research topic proposed within the ERCIM CoreGRID Working Group

Title	Advanced programming models for Services
Proposer	M. Danelutto, Dept. Computer Science, Univ. of Pisa
Keywords	programming models, structured programming, group communications, skeletons, design patterns, distributed programming abstractions, service orchestration, service choreography, interactive and real time services, HPC services, autonomic computing, self-* concerns, QoS, SLA, grid and cloud architectures

Research topic proposal

Recent advances in software service technology increasingly provide interoperable frameworks and tools suitable for development of new applications/services from existing ones. Unfortunately, however, the level of abstraction presented to the service developer is currently not very high. New paradigms are necessary to raise the level of abstraction presented to service users, particularly in the following areas:

- **Advanced Programming Models for Services.** The effort needed to combine existing services to develop a new service/application, should be made less onerous and less costly by providing new paradigms which offer the service user/developer new and effective means to operate on/with services, means that should be far from the low level, machine-oriented service mechanisms currently used.
- **Autonomic Management of Service Applications.** Support for the development of autonomic service managers to take care of self-* (non-functional) properties should be provided through abstractions that may be instantiated, generalized or specialized by service users to implement autonomic service management.

Both these areas require substantial investigation, and, where appropriate, reuse of existing results in other research areas, together with the development of new, service-oriented mechanisms, tools, programming models and frameworks. The related research challenges can be summarized as follows:

1. Identify appropriate, high-level, reusable and parametric programming patterns to be used in the design and implementation of applications built from service orchestration;
2. Investigate/design high-level, general purpose, possibly unconventional programming model(s) supporting both workflow-like and more general and more coupled distributed/parallel programming patterns on top of services and service orchestrations;
3. Identify appropriate, high-level, reusable and parametric programming patterns to be used to implement individual autonomic managers taking care of the self-* concerns in service applications;
4. design suitable policies to support coordinated (e.g. hierarchical) as well as multi-goal autonomic management taking care of self-* concerns in complex service applications.

In more detail, the research activities of this research topic will cover areas such as:

- high-level programming paradigms for services, possibly inheriting results from the component and the skeleton/design pattern research frameworks;
- paradigms typical of real-time, interactive services, which are becoming increasingly important and challenging;