A Hierarchical Approach for Resource Management in Heterogeneous Systems

HEAP LAB
POLITECNICO MILANO 1863

Michele Zanella, Federico Reghenzani
Giuseppe Massari, William Fornaciari
{name.surname}@polimi.it

OUR RESEARCH

Heterogeneous architectures are emerging as a dominant trend for HPC, mainly thanks to their high performance-per-watt ratio.

Dealing with heterogeneity and task-based applications requires to consider different aspects at both infrastructures level and single node in order to meet power, thermal and performance requirements. Thus, in order to provide an effective and fine-grained management of the available resources, as well as balancing the load by dispatching applications among the different computing nodes, we proposed a hierarchical approach in which different resource managers, running in the nodes, collaborate to reach a multi-objectives optimization. In particular, we need to tackle different challenges: (a) dealing with task-based applications in which each task may have data and timing dependencies with other ones; (b) the execution of taskon different heterogeneous computing units could lead to obtain different QoS or power consumption; (c) applications' requirements must be satisfied, while addressing system'sconstraints (power/thermal/energy); d) the infrastructure is composed by various computational nodes equipped with heterogeneous resources.

INFRASTRUCTURE-LEVEL MANAGEMENT

At the top of the hierarchy we need to consider the infrastructure-wide requirements and constraints.

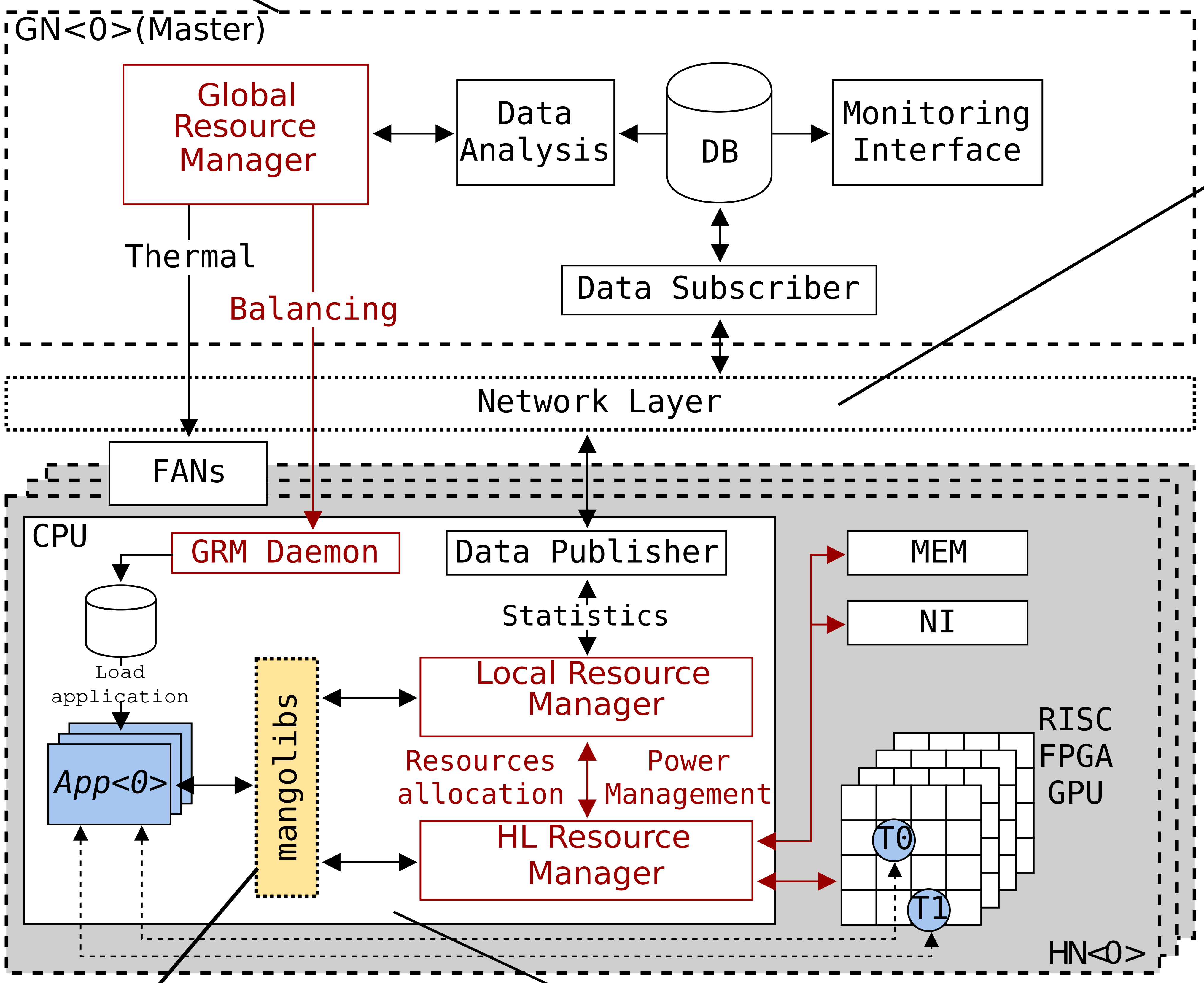
At infrastructure-level we need to achieve two objectives:

(1) performing load balancing among the slave nodes;

(2) performing an effective thermal management.

Load balancing is important to maintain a good QoS, preserving an acceptable power and thermal conditions.

The GRM [4] is in charge of dispatching the execution of an application on a specific node, basing on the global status of the architecture. The latter comprises the information collected from the LRMs, enriched by thermal and cooling information. In fact, thermal management needs to be addressed jointly with cooling control to ensure reliability and maximize energy efficiency. Thus, allocation strategies can leverage on those information to exploit the new architectures and the heterogeneity of the entire platform for the particular target applications.



PUBLISH/SUBSCRIBE DATA LAYER

In order to provide the aforementioned management, a Data Layer is introduced to collect run-time statistics coming from the LRMs and to provide them to different actors, such as Data Analytics software and Monitoring interfaces.

In order to be scalable and easily extensible, the Data Layer is based on a publish-subscribe mechanism.

The master node exploits a Data Communication Interface implementing a Data Subscriber in order to subscribe to different information related to specific slave nodes. In turn, each slave node's LRM has been extended with a Data Publisher that collects subscriptions possibly coming from different subscribers and push to them the most updated required information.

RUN-TIME LIBRARY

In the MANGO infrastructure, the applications must be written with the dedicated programming model [1], similar to OpenCL. Each host-side executable is linked with the run-time library called libmango. The library is in charge of exchanging the run-time information with the LRM BarbequeRTRM [2].

The application provides to the library the description of the workload to be deployed on heterogeneous node as a task graph. The task graph is a direct, possibly cyclic, graph, with the arcs representing the mapping between the kernel, the buffers and the events. The nodes can be buffers or kernels and two adjacent nodes cannot be of the same type. The resource manager exploits the task graph to pick the best resource allocations among the available ones.

LOCAL NODE MANAGEMENT

The resource assignment is a complex decision that has to consider:

(1) the available processor types and the target architectures of the application kernels;

(2) the number of threads of the kernels and the number of cores of the processors;

(3) the NoC resources, in terms of bandwidth and availability of virtual networks;

(4) the memory allocation, especially considering the internal fragmentation;

(5) the temperature and power requirements at different layers, from the single core to the entire infrastructure;

(6) the potential cache conflicts and optimizations;

(7) the application QoS and its variability.

Taking in account all of these factors entails a multi-dimensional and large decision space, especially in the presence of multiple applications. For this reason, the decision is split among different actors: the HRM is in charge of verifying the availability of low-level resources, such as the NoC bandwidth.

Given the feasible allocations, the local resource manager explores the still large decision space helped by a dedicated memory manager.

REFERENCES

[1] G.Agosta, W. Fornaciari, G. Massari, A. Pupykina, F. Reghenzani, and M. Zanella.
Managing heterogeneous resources in hpc systems.
In Proceedings of the 9th Workshop and 7th Workshop on Parallel Programming and RunTime Management Techniques for Manycore ARchitectures and Design Tools and Architectures for Multicore Embedded Computing Platforms, PARMA-DITAM '18, pages 7-12, 2018.

[2] P. Bellasi, G. Massari, and W. Fornaciari.
Effective runtime resource management using linux control groups with the BarbequeRTRM Framework.
ACM Trans. Embed. Comput. Syst, 2015

[3] J. Flich, G. Agosta, P. Ampletzer, D. A. Alonso, C. Brandolese, A. Cilardo, W. Fornaciari, Y. Hoor-nenborg, M. Kovac, B. Maitre, G. Massari, H. Mlinaric, E. Papastefanakis, F. Roudet, R. Tornero, and D. Zoni.
Enabling hpc for qos-sensitive applications: The mango approach.
In 2016 Design, Automation Test in Europe Conference

[4] A. B. Yoo, M. A.Jette, and M. Grondona..
Slurm: Simple linux utility for resource management..
In Dror Feitelson, Larry Rudolph, and Uwe Schwiegelshohn, editors,Job Scheduling Strategies forParallel Processing, 2003

FUTURE WORKS

We plan to prove the benefits of our approach inside the MANGO project context [3] and to apply it also in emerging computing scenarios, such as Distributed Mobile Computing and Fog paradigms, where heterogeneous resources are spread across different interconnected devices.

Current ongoing works are trying to solve the resource allocation problem using genetic algorithms to find a quasi-optimal allocation.