Akka-based Support for Multiscale Applications

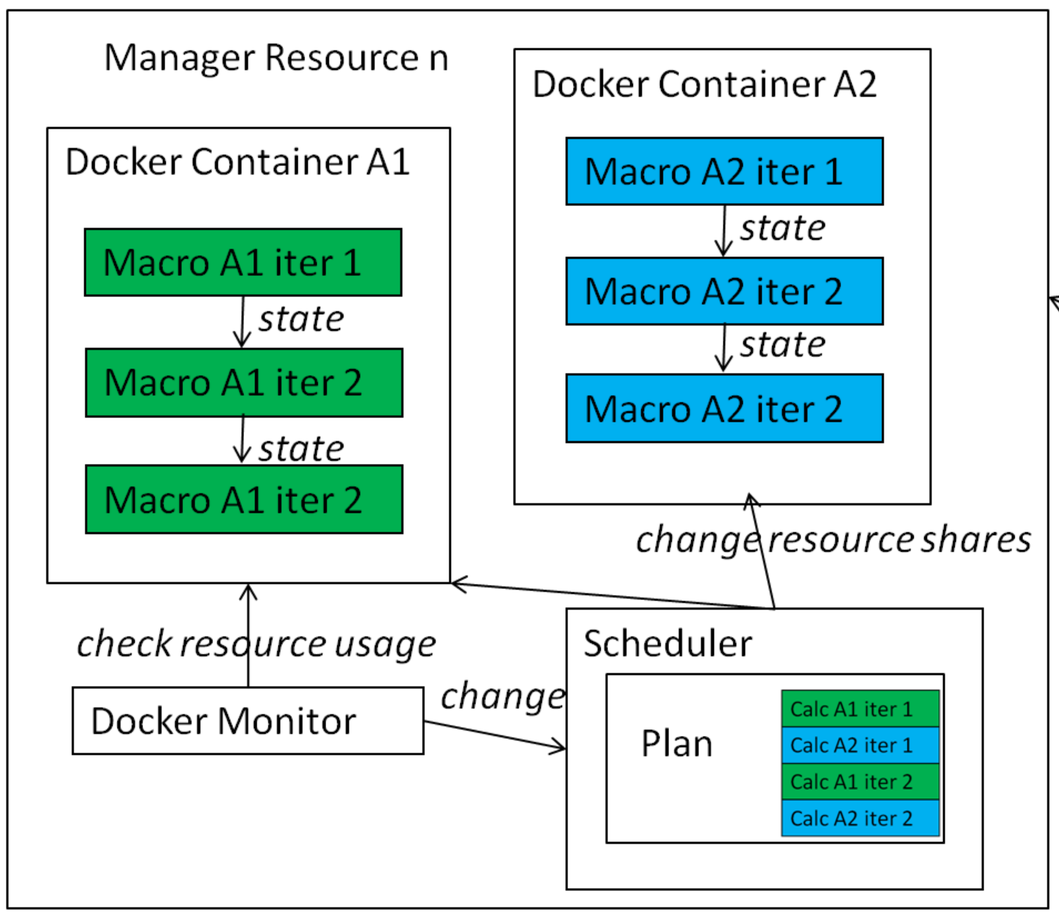
Katarzyna Rycerz and Marian Bubak
AGH University of Science and Technology, Department of Computer Science, al. Mickiewicza 30, 30-059 Kraków, Poland
ACC Cyfronet AGH, Nawojki 11, 30-950,Kraków, Poland

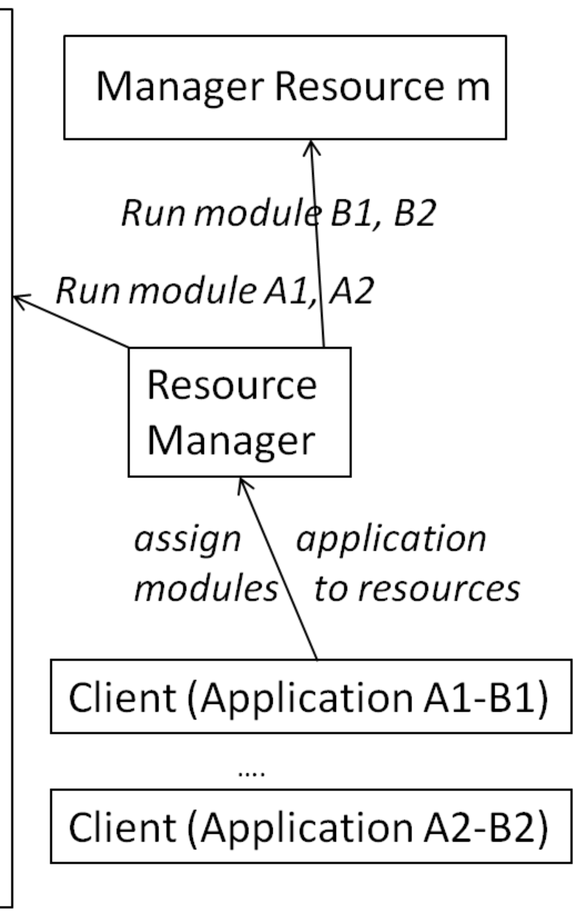
Motivation

- Different coupling templates of multiscale applications [1] supported by building and execution tools [2,3]
- Efficient execution support for iteration graphs of multiscale applications on a fine-grained (iteration level)

Solution

- Resource Manager groups modules of similar demands of resources
- modules on the same resources may belong to different applications
- Docker containers are used to separate environments of each module
- the system monitors containers resources usage
- Scheduler changes shares of the resources between containers to realize the plan: one module is executed in the idle time of the others
- the Managers, Monitor and Scheduler and implemented as Akka actors
- the communication of Managers and Clients with the Spray toolkit

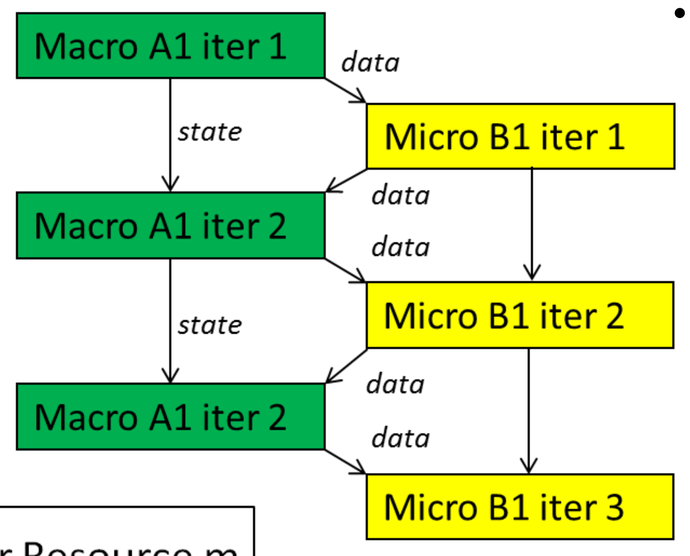




Typical execution cases of macro-micro interactions

Case 1

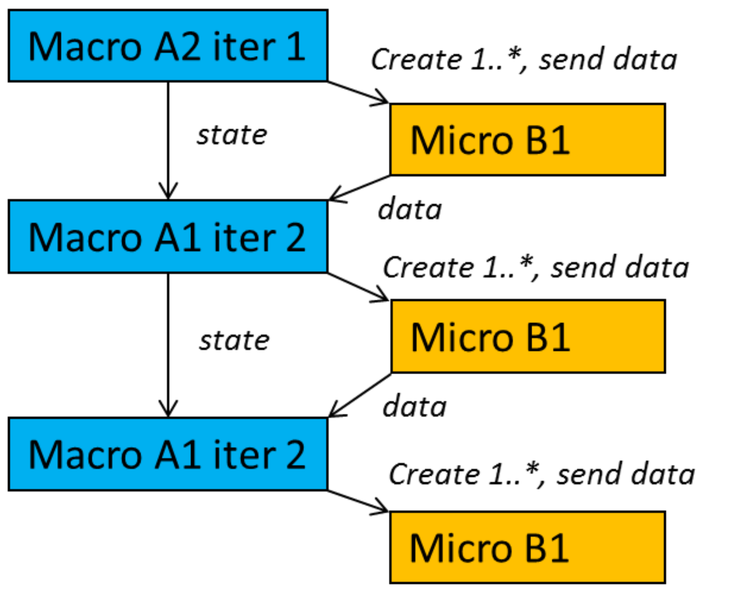
- macro and micro modules execute concurrently
- state from iteration *i* is then used in iteration *i+1*.

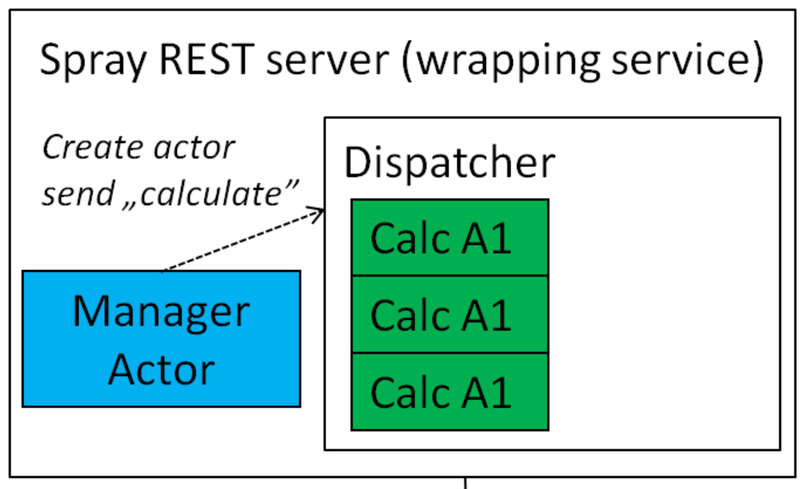


- if the macro module needs to use different resources then micro module it has to wait wasting resources.
- we propose a solution that allows for efficient use of resources during these idle times.

Case 2

In each iteration macro model creates several micro modules, sends data to them and waits for the results





Experiment Results

- computing is invoked by communicating with the wrapping service by REST (Spray toolkit)
- the manager actor inside a wrapping service creates a single calculation actor for one CPU intensive calculation.
- a single calculation is an LU factorization algorithm with parameterized matrix size to steer CPU time of the simulation

Number of calculation tasks	Task creation time	σ
1	980	14
10	1064	7
20	1047	15
30	1100	8
40	1131	15
100	1272	20
1000	3280	20
10000	14444	125

Tab 2. Akka and Spray overhead measured as a time of tasks creation by a REST client

- LU factorization was implemented with LAPACK library routine wrapped by Ruby linalg library
- Akka 2.3.11 for Scala 2.11 on Intel Core i7-2600 CPU @3.40 GHz.

Tasks nb	Naive method	σ	Akka method	σ
1	1.56	0.05	1.56	0.05
10	10.7	0.4		
20	21.5	0.7		
30	32.4	0.5		
40	43.6	0.5		
50	98	8		
100	n/a	n/a		
1000	n/a	n/a		
10000	n/a	n/a		

Tab 1. Execution of the varied number of calculations on the same CPU resource by running them concurrently (naive method) and one after another with Akka-based queuing mechanism (customized dispatcher with thread-pool-executor)

Conclusions

- The legacy multiscale applications[4] are supported by enclosing them in Docker environment
- In the future, as a resources, apart from HPC we also plan to use cloud resources and mechanism of lightweight virtualization containers

1. Joris Borgdorff, at al: Foundations of distributed multiscale computing: Formalization, specification, and analysis, JPDC, Volume 73, Issue 4, April 2013, Pages 465-483
2. E. Ciepiela at al. Exploratory Programming in the Virtual Laboratory, in Proceedings of the International Multiconference on Computer Science and Information Technology pp. 621–628[3]
3. K. Rycerz, et al: An Environment for Programming and Execution of Multiscale Applications, Future Generation Computer Systems, 2015 vol. 53, s. 77–87.
4. K. Rycerz, et al: Enabling Multiscale Fusion Simulations on Distributed Computing Resources. In: M. Bubak, J. Kitowski, K. Wiatr (Eds.): eScience on Distributed Computing Infrastructure, LNCS, Vol. 8500. Springer, pp. 195-210 (2014)

This study was partly supported by the AGH grant no 11.11.230.124 and also by the European Union within the European Regional Development Fund program no. POIG.02.03.00-12-137/13 as part of the PLGrid Core project

INNOVATIVE ECONOMY
NATIONAL COHESION STRATEGY

The work was co-funded by the European Regional Development Fund as part of the Innovative Economy program.

http://dice.cyfronet.pl

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND