



# SymGrid-Services

A framework for dynamic composition  
of symbolic services

Georgiana Macariu

Marc Frîncu

Alexandru Cârstea

Prof. Dana Petcu - Coordinator

science@ieat.ro

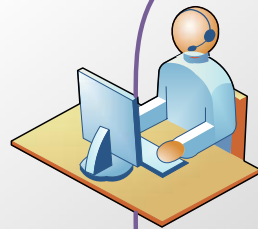


This research is supported by SCIENCE, a European project under the Seventh Framework Programme (FP7) for research and technological development

# SymGrid-Services

- ★ access Grid and Web services from a Computer Algebra System (CAS)

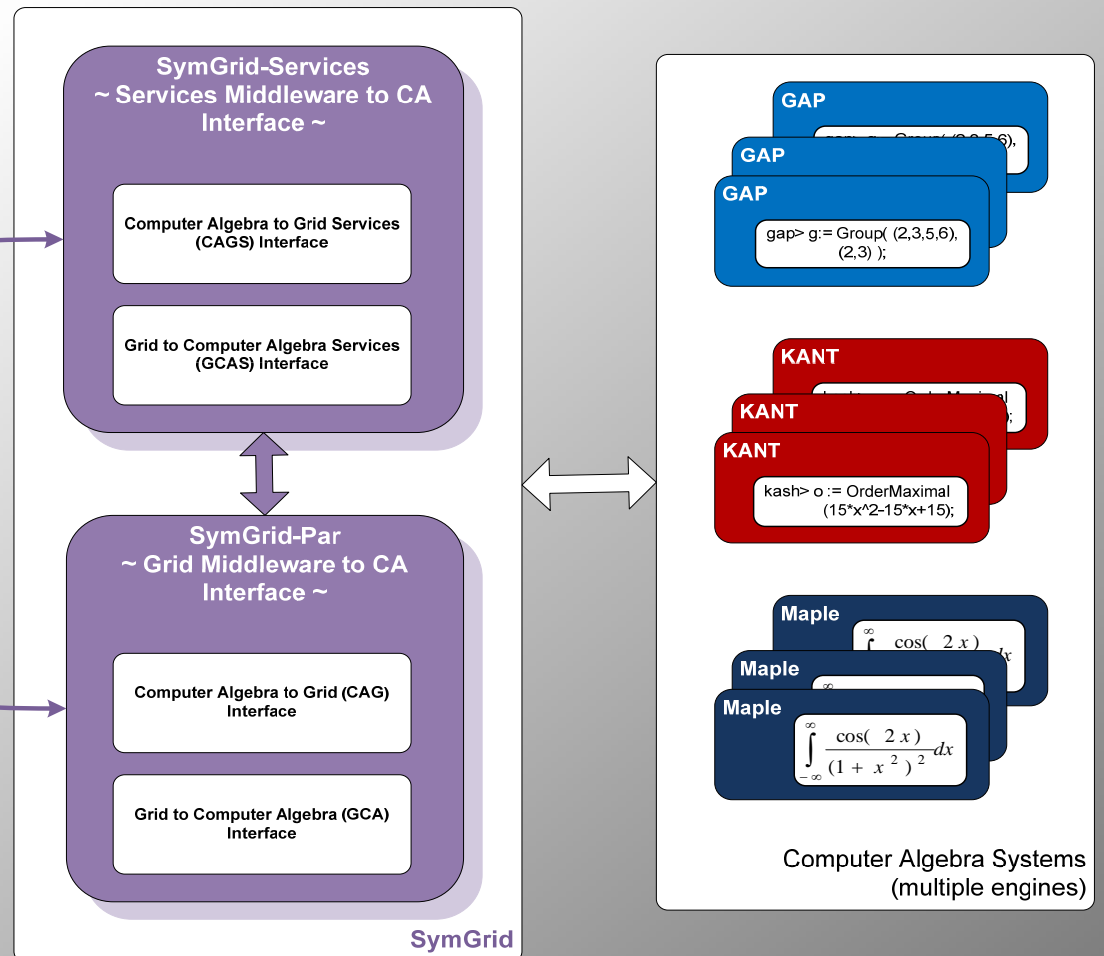
- ★ compose CAS' functionalities deployed as Web services



- ★ proof-of-concept: GAP package available

- ★ available at:

<http://science.ieat.ro>





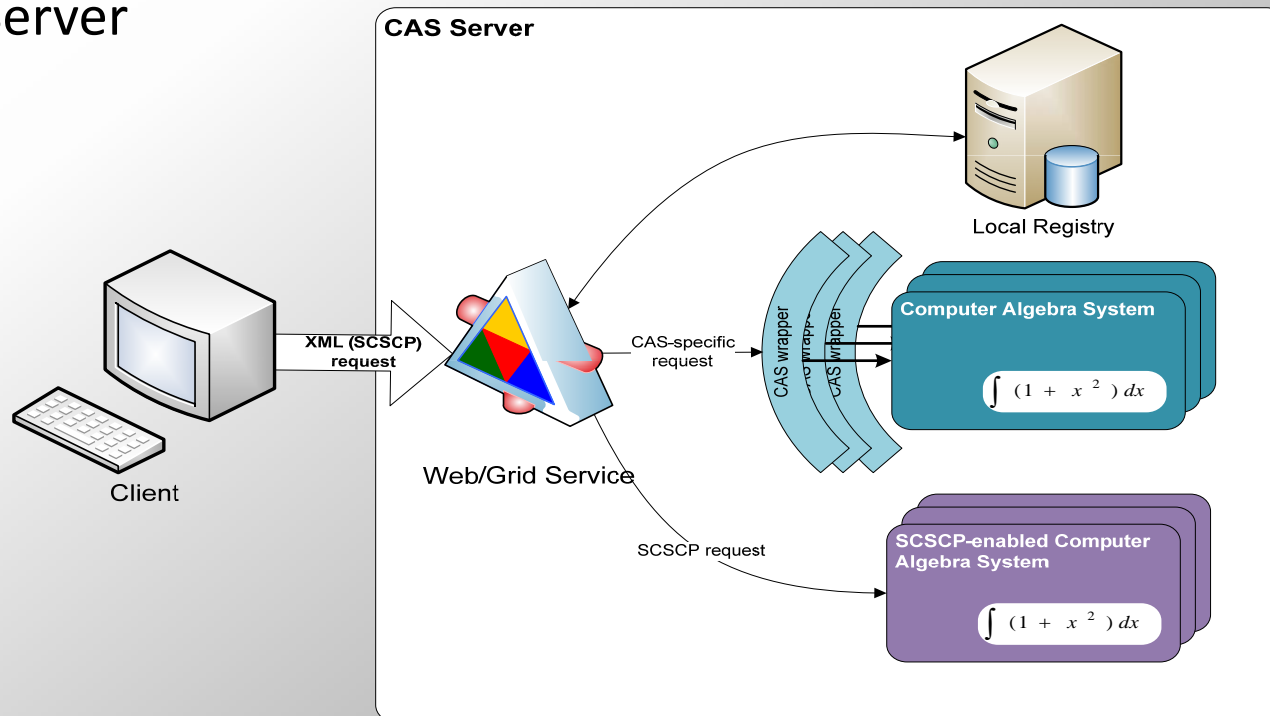
# Computer Algebra Systems

- ★ Computer Algebra Systems(CAS) are the main software packages used for symbolic computations
- ★ Symbolic computing main limitations:
  - ☆ lack of sufficient memory resources
  - ☆ long running tasks
- ★ CASs may be both clients and servers if they can be integrated in distributed architectures
- ★ CAS lack of support for integration within computer networks
- ★ General solutions for legacy software integration:
  - ☆ Redesigning applications
  - ☆ Integration using wrapper components

# CASs as servers

## ★ CAS Server components:

- ★ Web service: the interface of the system with the exterior world
- ★ Wrapper: negotiates message exchange
- ★ CAS engine: execute the symbolic call
- ★ Local Registry: stores information regarding the state of the CAS Server





# The Web/Grid Service interface

★ The standard interface of CAS Servers offers:

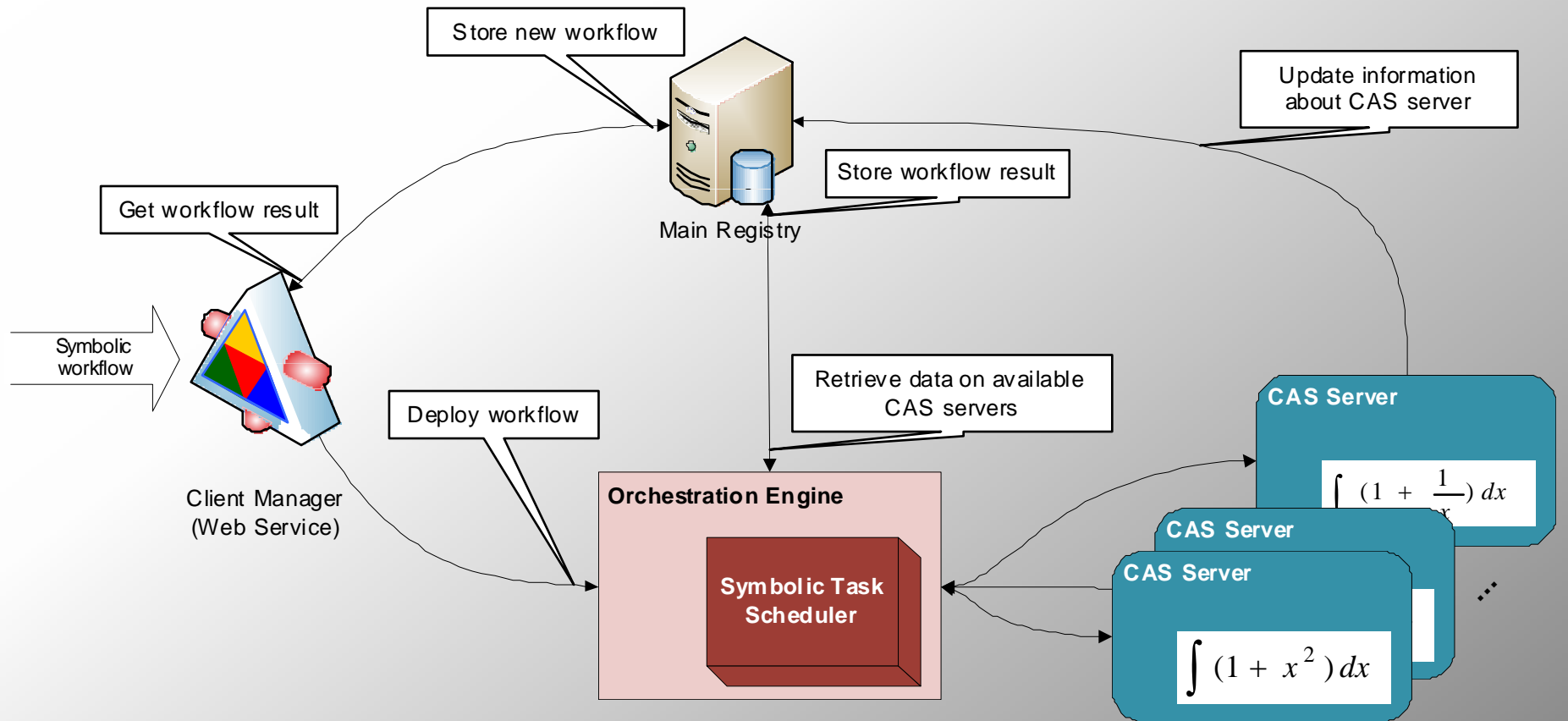
☆ an operation that receives the computation call (OpenMath object)

☆ callback functionality

☆ management services: pause/resume/cancel + set result

☆ discovery information sent to remote registries that act as centralized discovery registries

# The AGSSO component



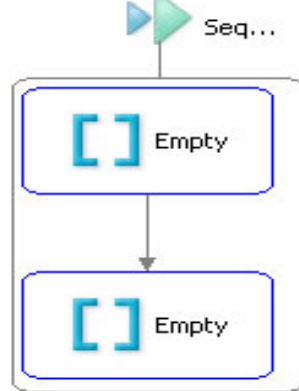


# Scientific workflows wish-list

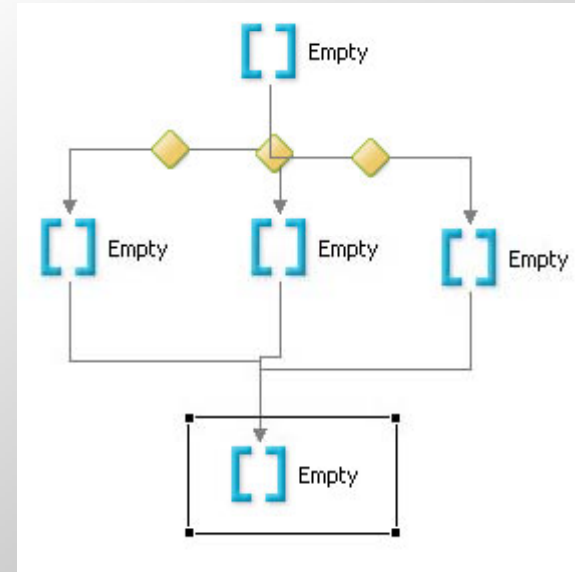
- ★ Computations may have thousands of steps
- ★ Need for tools that orchestrate the steps of scientific discovery
- ★ Reproducibility: requests meta information about the execution of the workflow
- ★ Researchers should find it easy to set up and execute
- ★ Most scientific activity consists of exploration of variants and experimentation with alternative settings: the need for start/stop/resume + steering
- ★ Inspect status and values obtained on the fly

# Workflow patterns

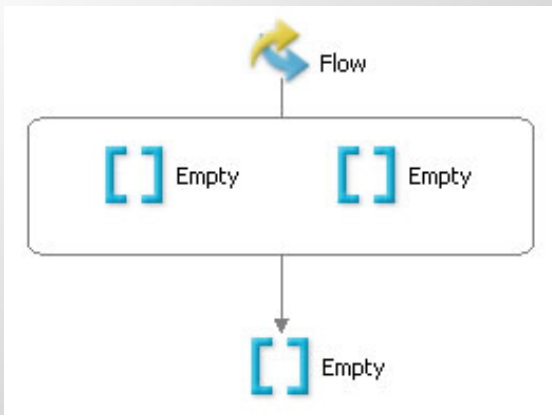
Sequence



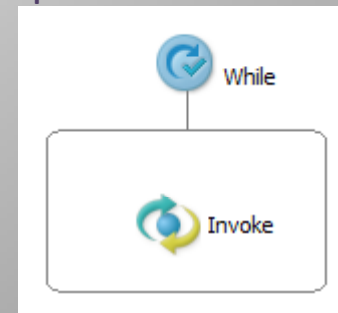
Conditional execution



Parallel



Repetitive execution





# Symbolic Computation Patterns

- ★ Apply In-place - apply a certain transformation on all the objects of a list
- ★ Apply New - create a new list based on the transformed objects of a given list
- ★ Count - count the objects in a list having certain characteristics
- ★ For Any - check if all objects in the list have certain characteristics
- ★ Fold - calculate a global value based on the elements of the list