Interface for a Computational Cluster: Resource Description Approach

Bogdanov A.
Ivanov P.
Gaiduchok V.
Kamande M.
Cubahiro A.

GRID'2016

Agenda

Introduction

Users

Administrators

Applications

Tests

Proposed approach

Conclusions

Conclusions

Introduction

- Creation of a convenient and simple user interface could be considered one of the most important tasks of an average computational center.
- Thought-out interface could improve user experience and resource utilization.
- Inexperienced users could spend much time to learh the interface.
- They could do mistakes which could lead to imbalance in system utilization.

Introduction

- Typical computational center consist of several clusters with different performance.
- Typical organization provide users with a variety of software applications, free and open source, as well as proprietary.
- Typical user is interested in one or two applications.

Users

- Users do not want to go deep into details of the system.
- They want the system interface to be simple.
- They need to calculate their tasks fast.
- They assume administrators will solve all problems.

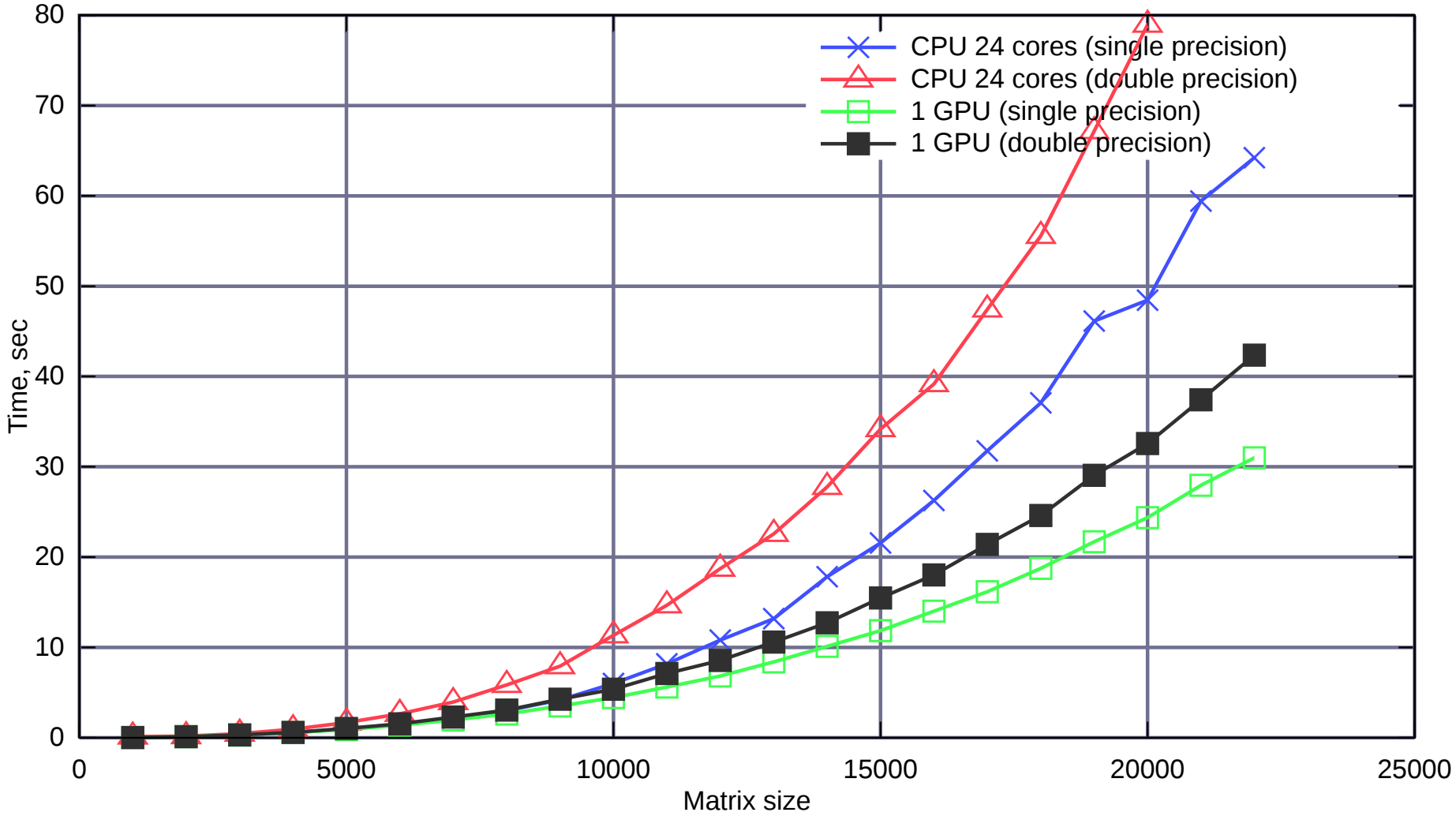
Administrators

- Administrators need to restrict users for different reasons (e.g. security).
- They need to comply with organization policies and to assure that users do it.
- They could improve user decisions in terms of performance of applications.
- But often it is hard to do if users have direct access to the underlying system.

Applications

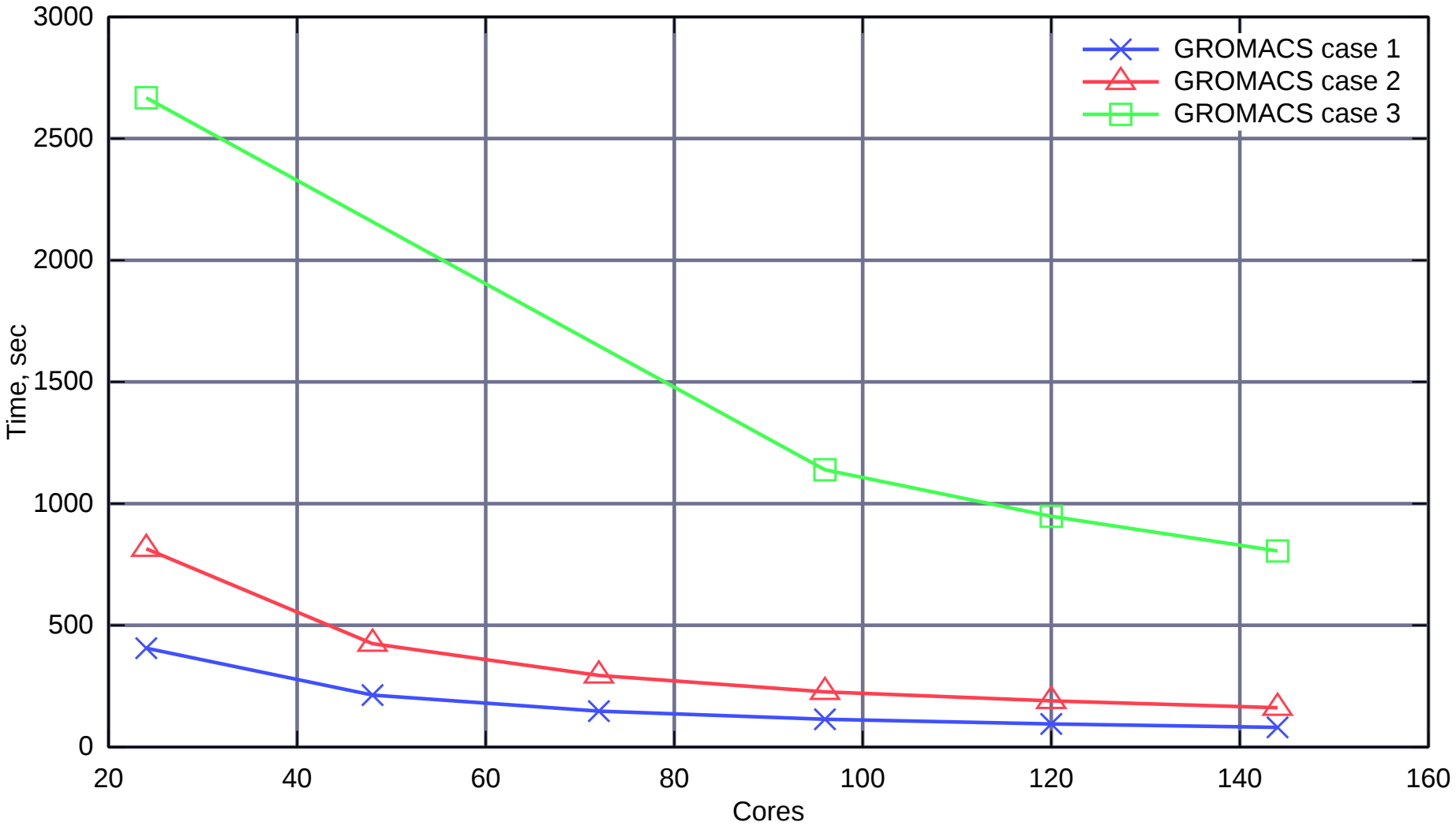
- Different applications have different scalability.
- They can use different libraries which have different performance.
- Tasks of the same application can use different modules and vary greatly in requirements for computational hardware.
- Limited scalability could lead to wasting hardware resources.
- It is difficult in general case to estimante the computation time and scalability of an average application.

Tests



Applications tests

Tests



Application tests

Proposed approach

- Traditional approaches usually delegates all the responsibility to users.
- They usually require specification of all the parameters of a task (harwdare requirements, system requirements, libraries, etc.).
- User is responsible for the correct choice of various parameteres which could be obscure.
- They often provide users with CLI only (e.g. generic PBS implementation).
- As a result some users should learn new sophisticated interfaces and go deep into details of the system while others just do many mistakes (in terms of task submitting, performance, etc.).

Proposed approach

- Description based system.
- Everything related to application should be written in a simple language.
- Each supported application gets its own description.
- The system provide administrators with a powerful language to describe the application, impose some limitations and improve performance.
- Users are given with a simple and intuitive GUI.
- Cross-platform: it is written in Java (JDK 1.8 is required) .

Proposed approach: administrators

- Administrators describe applications:
  - They choose environment variables (PATH, LD_LIBRARY_PATH, etc.), resource specification (default amount of resources, preferred node types, etc.), system limitations, etc.
  - They could e.g. prefer some MPI library.
  - Once it is done, it is available to all users, no additional manual configuration is required (e.g. changing ~/.bash_profile or ~/.bashrc).
- Such descriptions are propagated to all users.
- Administrators could impose some limitations (e.g. maximum number of cores for some application with bad scalability) in order to improve overall performance.

Proposed approach: applications

- One could write many descriptions with default settings: for a cluster, for a platform, for some particular node, for some library, etc.
- These specifications could be combined in order to make the administration easier.
- Descriptions could vary from specifying one description for some class of applications to some particular version of an application.
- Descriptions could be reused in order to facilitate the administration when one has to support many applications.

Proposed approach: users

- Users are provided with a convenient GUI.
- They choose resource (e.g. some particular cluster), application, version.
- All the necessary configurations are already described by administrators, so users could focus on their tasks.
- They could specify only the required parameters (e.g. input file with the task) and use the default settings proposed by administrators.

Example

default cores 4

...

warning *"[Algorithm] has bad scalability ..."*

...

option "-a" *"Algorithm to use"* "string" "REQ" ...
option "-s" *"Number of domains."* "int" "REQ" ...
option "-i" *"Input file"* "string" "REQ" ...

...

env PATH=/openmpi/1.6/bin:...
env LD_LIBRARY_PATH=/openmpi/1.6/lib:...

...

Comparison

- CLI is considered to be inconvenient by the vast majority of users.
- Direct access to underlying management system is considered to be a bad option by many administrators.
- Administrators often write scripts to ease task submission, but such approach requires CLI anyway.
- Existing (even web-based) GUI interfaces usually implies scripting: users work with GUI, but have to write scripts to submit jobs.
- There are paid proprietary solutions, but such solutions usually have many unnecesary features and are too expensive.

Conclusions

- Similar systems usually facilitate the access to the system (GUI instead of CLI), but not the work with the system (users are responsible to write scripts, etc.) .
- The main element is a specific application build descibed by administrators (invisible to users).
- Users will work with only really necessary aspects.
- The proposed approach meets the user requirements (simple and convenient way to calculate a particular task), as well as administrator needs (impose organization policies and limit the access to the underlying system).

Conclusions

- Such approach could improve resource utilization.
- Proposed system is designed to work with a generic PBS implementation.
- Web-interface is under development.

Questions

Thank you!