

Integration of Programming Models and Persistent Storage Systems



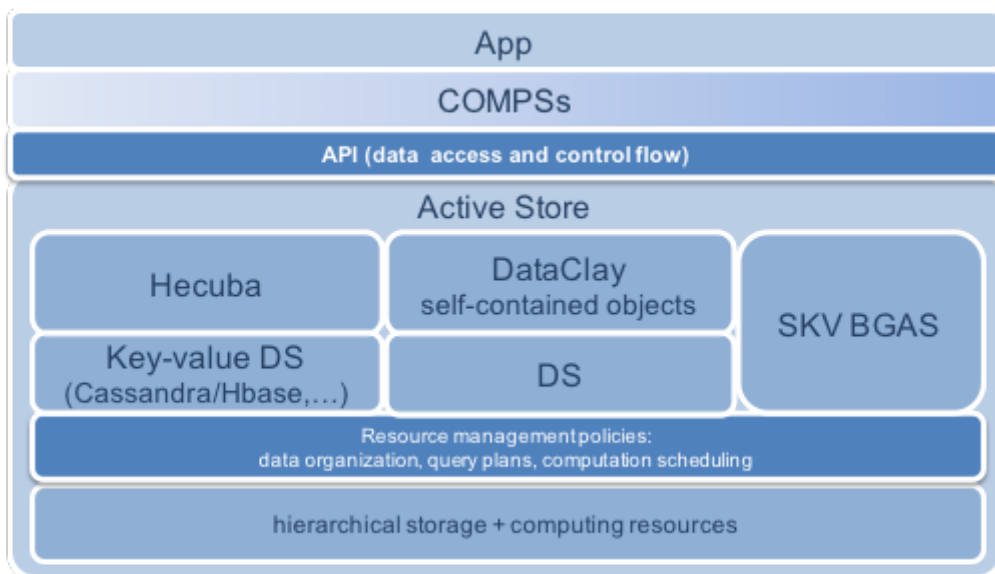
This research line is focused on the integration of [COMPSs](#) programming model with persistent storage systems in order to target Big Data and persistency problems.

Summary

In order to make [COMPSs](#) capable of orchestrating applications that process big amounts of data, the data management is being extended both at programming model and runtime system levels. In this sense, we are aimed at working on integrating COMPSs with persistent object storage platforms, making the most of distributed, fault-tolerant and efficient storage possibilities.

The main mission of the aforementioned platforms is to facilitate the access to data as much as possible. So that any function in a COMPSs application should be able to seamlessly work with objects in memory or with persistent objects, with no need to be adapted for each case.

The following picture depicts the overall structure of the current architecture. COMPSs applications are located at the top of the stack, and the tasks they generate are processed by the COMPSs runtime system. COMPSs runtime relies on a Storage API in order to create, delete, insert, retrieve and iterate over persistent data.



The Storage API can be implemented by multiple Storage Backends. The Storage Backend is responsible for storing data in a set of distributed resources, managing data format and organisation and optimising data queries. Currently, the Storage API does not provide any functions to manage transactions, so the consistency guarantees offered to applications are inherited from the Storage Backend used.

Our motivations are to make possible to handle big objects (too big to fit in the memory of a single node), offer a simple mechanism to access those objects and manage their persistency, and keep all data management transparent to the developer.

Objectives

- Do research and development for providing support for multiple storage technologies from the COMPSs environment. Specifically, the objective is to hide the complexities of storage technologies to users and exploit the benefits provided from the use of them, such as DataClay and Hecuba.
- Do research of new and emerging storage technologies.
- Design and provide a uniform API for interacting with storage technologies silently.
- Extend the support to other storage technologies.
- Do research and development in efficient scheduling algorithms, exploiting the metrics and data location information that each storage technology is able to provide in order to enhance the performance of users applications.

Barcelona Supercomputing Center - Centro Nacional de Supercomputación

Source URL (retrieved on 23 Abr 2024 - 11:11): <https://www.bsc.es/es/research-development/research-areas/big-data/integration-programming-models-and-persistent-storage>