dataClay

dataClay is a distributed object-oriented data store that enables programmers to handle persistence using the same model they use in their object-oriented applications, thus avoiding time-consuming transformations between persistent and non-persistent data models.
In addition, dataClay enables the execution of code next to the data. By moving computation close to the data, dataClay reduces the amount and size of data transfers between the application and the data store, thus improving performance of applications.

Key features

- **Single data model**: dataClay manages persistent objects using the same abstractions than the programming language, avoiding the time-consuming task of writing mapping code between the program's form of data and the one used by DBMSs or files.
- **Distribution**: in dataClay data can be distributed among several backends to provide scalability in the amount of data that can be handled, and to exploit parallelism in data-intensive applications.
- **Computation close to data**: dataClay stores object methods together with the data, thus being able to execute them in the backed where the data resides, instead of moving the data to the application address space.
- **In-memory data store**: dataClay exploits memory usage as much as possible to improve performance of client applications by keeping objects, and the references between them, instantiated as native language objects ready to be used.
- **Replica management**: dataClay offers a simple, customizable, and fine-grained mechanism to synchronize replicas in different backends. It enables the application of different synchronization policies (or none at all) depending on the type of data, thus paying the overhead of synchronization only when it is required by the applications.
- **Memory and disk management**: dataClay takes care of the unreachable objects that may be generated by applications by means of a garbage collector that frees the space they take both from memory and from disk.
- **Integration with COMPSs**: dataClay is fully integrated with the COMPSs parallel programming model and runtime, thus easing the development of applications that take advantage of data distribution and data locality.
- **Edge-to-cloud data store**: independent dataClay instances can be federated in order to build a shared object space among different machines, ranging from constrained devices such as Raspberry Pi or Jetson boards to HPC clusters or cloud datacenters.

Citing dataClay

Please, use the following reference when citing dataClay in your publications:

- dataClay: A distributed data store for effective inter-player data sharing. Jonathan Martí, Anna Queralt, Daniel Gasull, Alex Barceló, Juan José Costa, Toni Cortes. Journal of Systems and Software 131: 129-145 (2017), DOI: 10.1016/j.jss.2017.05.080

Acknowledgments

This project is supported by the Spanish Government through Programa Severo Ochoa (SEV-2011-0067, SEV-2015-0493), the Spanish Ministry of Science and Innovation (TIN2012-34557, TIN2015-65316, PID2019-107255GB, MCIN/AEI/10.13039/501100011033), the Generalitat de Catalunya (2009-SGR-980, 2014-SGR-1051, 2017-SGR-1414) the European Union's Horizon 2020 research and innovation program under projects mF2C (730929), NextGenIO (671591), BigStorage (642963), EXPERTISE (721865), ELASTIC (825473), eFlows4HPC (EuroHPC JU 955558), ADMIRE (EuroHPC JU 956748), and Horizon Europe program under project ICOS (101070177).

Barcelona Supercomputing Center - Centro Nacional de Supercomputación

---

Source URL (retrieved on *29 Mar 2024 - 12:47*): https://www.bsc.es/es/research-and-development/software-and-apps/software-list/dataclay