BSC releases COMPSs version 3.1 and releases dislib version 0.8.0

The Barcelona Supercomputing Center offers COMPSs to the HPC community, a set of tools that helps developers efficiently program and execute their applications on distributed computational infrastructures.

This COMPSs release includes several new features such as a new decorator and corresponding runtime extensions to support automatic data transformations before tasks' execution.

The Python binding comes with a new Julia decorator to support the integration of Julia in PyCOMPSs workflows.

dislib new release includes GPU support for some methods, new parallel methods (TSQR, KNN), and a generalization of the grid search for any type of simulations.

The Workflows and Distributed Computing team at the Barcelona Supercomputing Center-Centro Nacional de Supercomputación (BSC-CNS) is proud to announce a new release, version 3.1 (codename Margarita), of the programming environment COMPSs.

This version of COMPSs updates the result of the team’s work in the last years on the provision of a set of tools that helps developers to program and execute their applications efficiently on distributed computational infrastructures such as clusters, clouds and container managed platforms. COMPSs is a task-based programming model known for notably improving the performance of large-scale applications by automatically parallelizing their execution.

COMPSs has been available for the last years for the MareNostrum supercomputer and Spanish Supercomputing Network users, and it has been adopted in several recent research projects such as mF2C, CLASS, ExaQUte, ELASTIC, the BioExcel CoE, LANDSUPPORT, the EXPERTISE ETN, in the Edge Twins HPC FET Innovation Launchpad project and in sample use cases of the ChEESE CoE . In these projects, COMPSs has been applied to implement use cases provided by different communities across diverse disciplines as biomedicine, engineering, biodiversity, chemistry, astrophysics, financial, telecommunications, manufacturing and earth sciences. Currently, it is also under extension and adoption in applications in the projects AI-SPRINT, PerMedCoE, MEEP, CAELESTIS and DT-GEO. A special mention is the eFlows4HPC project coordinated by the group that aims to develop a workflow software stack where one of the main components is the PyCOMPSs/COMPSs environment.

The new release includes new features that helps on the integration of HPC, data analytics and Artificial Intelligence (AI) by providing a new decorator (@dt) to instruct the runtime to perform specific transformations to the tasks' data. Example of such transformation can be a change of the data organization from columns distributed in different nodes to rectangular blocks. This feature is exploited in the eFlows4HPC project to support the integration of HPC simulations with AI or data analytics functions. This new decorator is combined with the previously released @software decorator which has been enhanced with a new software description to allow tasks' parameters definition.

A new @julia decorator has been added to enable the combination of Julia tasks in PyCOMPSs workflows. Particularly, the tool Cobrexa used in the PerMedCoE projected is implemented in Julia. This extension would enable to develop workflows that include Cobrexa as a building block, with executions spanning one or multiple computing nodes.

Version 3.0 of COMPSs already included a mechanism to automatically record Data Provenance from the execution of COMPSs applications. This enables reproducibility and replicability of both COMPSs applications and dislib algorithms. Our approach is based on the RO-Crate 1.1 Specification that offers good integration to existing tools and frameworks. The mechanism has been extended in version 3.1 to Java COMPSs applications.

With the goal of extending the support to containerized environments, COMPSs can now work with uDocker for container tasks. This extension will be leveraged in the DT-GEO project where this containers' environment will be used.

COMPSs 3.1 comes with other minor new features, extensions and bug fixes.

COMPSs had around **1000 downloads** last year and is used by around **20 groups** in real applications. COMPSs has recently attracted interest from areas such as engineering, image recognition, genomics and seismology, where specific courses and dissemination actions have been performed.

The packages and the complete list of features are available in the Downloads page. A Docker image is also available to test the functionalities of COMPSs through a step-by-step tutorial that guides the user to develop and execute a set of example applications.

Additionally, a user guide and papers published in relevant conferences and journals are available.

For more information on COMPSs please visit our webpage: http://www.bsc.es/compss

DISLIB

The group is also proud to announce the new release of dislib 0.8.0. The Distributed Computing Library (dislib) provides distributed algorithms ready to use as a library. So far, dislib focuses on machine learning algorithms, and with an interface inspired by scikit-learn. The main objective of dislib is to facilitate the execution of big data analytics algorithms in distributed platforms, such as clusters, clouds, and supercomputers. Dislib has been implemented on top of PyCOMPSs programming model, Python binding of COMPSs.

Dislib is based on a distributed data structure, ds-array, that enables the parallel and distributed execution of the machine learning methods. The dislib library code is implemented as a PyCOMPSs application, where the different methods are annotated as PyCOMPSs tasks. At execution time, PyCOMPSs takes care of all the parallelization and data distribution aspects. However, the final dislib user code is unaware of the parallelization and distribution aspects, and is written as simple Python scripts, with an interface very similar to scikit-learn interface. Dislib includes methods for clustering, classification, regression, decomposition, model selection and data management. A research contract with FUJITSU had partially funded the dislib library and was used to evaluate the A64FX processor. Currently, the dislib developments are co-funded with 50% by the European Regional Development Fund under the framework of the ERFD Operative Programme for Catalunya 2014-2020, by the H2020 AI-Sprint project and by the EuroHPC eFlows4HPC project.

Since its recent creation, dislib has been applied in use cases of astrophysics (DBSCAN, with data of the GAIA mission), molecular dynamic workflows (Daura and PCA, BioExcel CoE). In the eFlows4HPC project, it is being applied in two use cases: in urgent computing for natural hazards (Random Forest regressors) and in digital twins for manufacturing (QR). In the AI-SPRINT project a personalized healthcare on atrial fibrillation detection is implemented using the Random Forest algorithm.

The new release 0.8.0 includes, for the first time, support for GPU based on CuPy for some methods: K-means, KNN, PCA, Matrix multiplication, Matrix Addition, Matrix Subtraction, QR factorization and Kronecker product.

In addition, a parallel implementation of the Tall and Skinny QR (TSQR) decomposition, an extension of the Grid Search to support the exploration parameters-sweep of HPC simulations, and a new KNN classifier have been added.

To better support the management of trained models, methods to load and save models have been added. In addition, other smaller operators and extensions to deal with the ds-array has been included.

Dislib 0.8.0 comes with other extensions and with a new user guide. The code is open source and available for download.

The Workflows and Distributed Computing group at the Barcelona Supercomputing Center aims to offer tools and mechanisms that enable the sharing, selection, and aggregation of a wide variety of geographically distributed computational resources in a transparent way. The research done in this team is based in the former expertise of the group, and extending it towards the aspects of distributed computing that can benefit from this expertise. The team at BSC has a strong focus on programming models and resource management and scheduling in distributed computing environments.

Barcelona Supercomputing Center - Centro Nacional de Supercomputación

Source URL (retrieved on *24 abr 2024 - 17:24*): https://www.bsc.es/ca/node/59943