ClusterControl: a web interface for distributing and monitoring bioinformatics applications on a Linux cluster

Gernot Stocker, Dietmar Rieder and Zlatko Trajanoski

Institute of Biomedical Engineering and Christian-Doppler-Laboratory for Genomics and Bioinformatics, Graz University of Technology, Krenngasse 37, 8010 Graz, Austria

Received on June 2, 2003; accepted on August 19, 2003
Advance Access publication January 29, 2004

ABSTRACT

Summary: ClusterControl is a web interface to simplify distributing and monitoring bioinformatics applications on Linux cluster systems. We have developed a modular concept that enables integration of command line oriented program into the application framework of ClusterControl. The systems facilitate integration of different applications accessed through one interface and executed on a distributed cluster system. The package is based on freely available technologies like Apache as web server, PHP as server-side scripting language and OpenPBS as queuing system and is available free of charge for academic and non-profit institutions.

Availability: http://genome.tugraz.at/Software/ClusterControl
Contact: zlatko.trajanoski@tugraz.at

INTRODUCTION

With the introduction of high-throughput technologies, e.g. sequencing and microarrays for expression profiling the amount of data that has to be stored, managed and analyzed increased dramatically. Very soon, it became necessary to implement computer-intensive bioinformatics applications for analyzing the flood of data on multi-processor computing systems. As most of these applications are not parallelized, they have to be distributed and monitored using a queuing system. Several web interfaces (Jenuth, 2000; Ferlanti et al., 1999; Stoesser et al., 2003; Blanchet et al., 2000) and more generalized approaches (Letondal, 2001) for universal generation of web interfaces based on textual xml descriptions have been developed previously to enable delivery of calculations on a computing server. However, since these web interfaces were developed for execution on the same server on which the web server is installed, they are not able to use efficiently the power of computing cluster systems.

The first approach to combine distributed calculations using a low-cost PC computing cluster and a web server has been implemented by the BeoBLAST (Grant et al., 2001) project. However, this system is limited to the BLAST (Altschul et al., 1990) applications and there is currently no freely available tool, which enables integration of other command line oriented programs like HMMer (Durbin et al., 1998) or FASTA (Deshpande et al., 1995). Therefore, the objective of this work was to develop a platform-independent web interface for distributing and monitoring bioinformatics applications on PC-based cluster systems.

PROGRAM OVERVIEW

The modular framework of this system enables integration of every command line-driven tool (Fig. 1). By adding a web-form (that includes all required parameters of the program which should be integrated) and an additional PHP file (that assembles the appropriate command line), the framework can be extended by any user with moderate programming skills. The delivery to the queuing system and collection of the results are managed by the framework in the background. Hereby, single process oriented applications are supported as well as real parallel applications programmed with special communication libraries like MessagePassingInterface (MPICH). After obtaining the results, it is also possible to run a post-processing tool to improve the visualization of the result files. An example for an integrated application of this framework is the existing NCBI-Blast module that can be tested on http://genome.tugraz.at/Software/ClusterControl/Demo. Currently, the list of supported application modules contains NCBI-Blast, Fasta, WU-Blast and HMMer. Customizations of existing modules can be easily done by using simple but well-defined interface classes and procedures. Unsupported applications can be added immediately and can extend the functionality of ClusterControl by following the short instructions on how to write new modules.

For time-consuming applications, the user can log out from the web interface and can login later to monitor the status of the calculation. If the calculation is still running, an automatically refreshing web page will show the current status...
Fig. 1. Schematic overview of ClusterControl. The user can submit jobs through the applications front-ends and monitor the current status of the cluster. After submission, the job will be distributed through the backend PHP classes and the OpenPBS queuing system on the cluster. The job also appears in the Results section and shows its current status. After calculation, a visual improvement of the results can be performed and the final result will be shown in the Results section.

of this submitted job. Otherwise, the processed result will be presented immediately to the user.

An additional feature of this web interface is the ability to monitor all cluster nodes attached to this cluster system. To provide the real-time status and functionality of every calculation node, a lightweight program is installed on every calculation node within the cluster. For the transfer of this status information to the web interface a text protocol was defined, which allows request for information, such as system-, processor-, memory-load, etc.

For the management of the users two different systems are supported: a Lightweight Data Application Protocol (LDAP)-Server or an Unix-like password file that can be used as source of user-accounts. Thus, local cluster user and virtual user
stored in the password file can access the cluster system simultaneously without having direct access to the data, sequences, etc. of each other.

The object-oriented application programming interface (API) of ClusterControl can also be used in PHP scripts for farming out program execution outside the web server context. Similar to the Mollusc-API (Hokamp et al., 2003), which is programmed in Perl and wraps up distributed program execution, the backend library of ClusterControl can be used to submit calculation jobs from a PHP script run in a regular Unix shell.

**INSTALLATION**

For local installation, ClusterControl requires a running installation of the web server Apache (http://www.apache.org) with PHP (Version 4) and LDAP support, that comes with nearly every Linux distribution and a local installation of the queuing system OpenPortableBatchSystem (PBS), which can be downloaded from www.openpbs.org. If an LDAP-Server is already available for user management of the cluster, it can be easily integrated by modifying the configuration file. To activate the monitoring mechanism of ClusterControl, a status collecting server-process must be started on every cluster node. With the adaptation of one well-documented configuration file the installation process is completed. This web application is freely available and can be downloaded from http://genome.tugraz.at/Software/ClusterControl, It is licensed under the Gnu General Public License (GPL; http://www.gnu.org/licenses/gpl.html).

**FUTURE DEVELOPMENT**

Future development of ClusterControl will provide extensions of the already existing application set, improvements in integration of features of OpenPBS for job management and extension to other queuing systems. Beside already supported MPI implementations, some parallel bioinformatics applications are still using the parallelization software PVM. Therefore, support for this platform is also in preparation. Additional enhancements in the form of new modules by the scientific user community are encouraged and will be integrated in the open source distribution of ClusterControl.

**ACKNOWLEDGEMENT**

This work was supported by a grant Bioinformatics Integration Network from the Austrian GEN-AU Programme.

**REFERENCES**