The DOTPAR Project: Towards a Framework Supporting Domain Oriented Tools for Parallel and Distributed Processing

José C. Cunha, Pedro Medeiros, João Lourenço, Vítor Duarte, João Vieira, Bruno Moscão, Daniel Pereira, and Rui Vaz

{jcc, pm, jml, vad }@di.fct.unl.pt
{jpdv, bmoscao, dlp, rpav}@asc.di.fct.unl.pt
Departamento de Informática, Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa, Portugal

Abstract. We discuss the problem of building domain oriented environments by a composition of heterogeneous application components and tools. We describe several individual tools that support such environments, namely a distributed monitoring and control tool (DAMS), a process-based distributed debugger (PDBG) and a heterogeneous interconnection model (PHIS). We discuss our experience with the development of a Problem Oriented Environment in the domain of genetic algorithms, obtained by a composition of heterogeneous tools and application components.

1 Introduction

We are developing a framework for the generation of problem domain oriented environments that can exploit parallel and distributed processing. In many distinct domains, parallel processing solutions are necessary, e.g. intensive simulation processes, but they must fit within heterogeneous environments also including visualization, interactive control, virtual reality interfaces, and access to large and complex databases. At a conceptual level this research is organized according to a hierarchy of levels including formalisms for software architectures [5], coordination models, resource management, interconnection models, and monitoring and control services. All along these levels there is a need to develop associated tools supporting the user in the specification and design, analysis, verification and evaluation activities. There are tools operating at the intracomponent level, as we find in current software engineering environments [6]. Other tools must operate at the intercomponent level, in order to support the process of application building, by selecting, evaluating and testing, configuring, activating, interconnecting, and monitoring and controlling the execution of multiple heterogeneous application components. In the following we discuss our experimentation with the design and implementation of tool prototypes and their integration in order to generate a problem domain oriented environment for the parallel execution of genetic algorithms.

2 Filling in Some of the Software Architecture Layers

The DAMS Monitoring and Control Layer The Distributed Applications Monitoring System is a flexible software layer supporting the monitoring and control of possibly heterogeneous parallel and distributed computations. A collection of distributed target application processes is monitored by DAMS system in order to gather status information, and control the execution. The DAMS architecture consists of a central Service Manager, a Local Manager on each node of the distributed architecture, and a Driver Process which is associated with each Target Process. Each service that DAMS is able to provide to a client tool must be supported by a distinct pair (Service Module, Driver Process). The Service Module takes care of the client tool functionality, and the Driver Process controls the direct interfacing to the Target Process.

A Distributed Debugging Tool PDBG is a distributed process-level debugging interface, supported as a DAMS service. It consists of one Debugging Service Module and a Driver Process for each target process being debugged, which is responsible for state inspection and control ([1][3][2]).

An Interconnection Model A heterogeneous application is built by interconnecting multiple components using the PHIS model which is based on a specialized form of process group. It provides a very general form of component interaction where the programmer must identify, in each component, the processes that will interact with the other components. The modifications made to each component consist of integrating each process into the relevant groups if multi-party component interaction is required, and the exchanging of interconnection messages ([4]).

3 A Genetic Algorithm Oriented Heteregeneous Environment

We have used some of the previously mentioned components in order to build a problem domain oriented environment for parallel execution of genetic algorithms (GA) in a heterogeneous environment. It has facilities for visualization, in real time, of the simulation evolution and for interactive steering, corresponding to the following components (see Fig. 1): a Genetic Algorithm (GA) component supports parallel genetic algorithm execution; a Data Visualizer (DV) visualizes the evolution of the GA simulation; and an Interactive Steering (IS) component supports the interactive steering of the simulation.

4 Conclusions and Current Status

The DOTPAR (Domain Oriented Tools for Parallel and Distributed Processing) Project aims to generate domain oriented environments out of existent component tools. This should be supported by associated tools handling abstractions

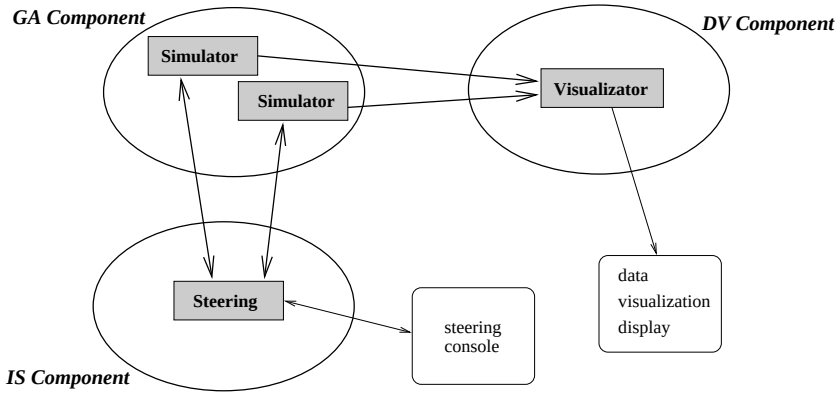

Fig. 1. GA Environment Software Architecture

at the inter-component level. Our experimental approach is incrementally giving us the knowledge to build such advanced tools. We have already developed some of the required models and tools (DAMS, PDBG, and PHIS), and are building specific environments such as the Genetic Algorithms case. A prototype exists with the described functionalities of parallel execution, visualization, and steering that can be tested by end-users.

Acknowledgments To B. Horta, L. Duarte, J. Duarte, N. Neves, G. Fert, EC COPERNICUS SEPP (CIPA-C193-0251) and HPCTI (CP-93-5383), Portuguese CINCIA and PRAXIS XXI PROLOPPE (3/3.1/TIT/24/94), Setna-ParComp (2/2.1/TIT/1557/95) and DEC EERP PADIPRO (P-005).

References

1. Cunha, J., and Loureno, J. An Experiment in Tool Integration: the DDBG Parallel and Distributed Debugger. *EUROMICRO Journal of Systems Architecture, 2^{nd} Special Issue on Tools and Environments for Parallel Processing* (1997).
2. Cunha, J. C., Loureno, J., Vieira, J., Mosco, B., and Pereira, D. A framework to support parallel and distributed debugging. Tech. rep., Departamento de Informtica, Universidade Nova de Lisboa, Apr. 1998. Proceedings of HPCN98.
3. Kacsuk, P., Cunha, J., Dzsa, G., Loureno, J., Fadgyas, T., and Anto, T. A Graphical Development and Debugging Environment for Parallel Programs. *Parallel Computing*, 22 (Feb. 1998), 1747–1770.
4. Medeiros, P. D., and Cunha, J. C. Interconnecting Multiple Heterogeneous Parallel Application Components. In *Proceedings of EuroPar'97* (Passau, Germany, Aug. 1997).
5. Shaw, M., et al. Abstractions and tools for software architectures and tools to support them. *IEEE Transactions on Software Engineering* (1995).
6. Winter, S., et al. Software Engineering for Parallel Processing, Copernicus Programme. Progress report 1, University of Westminster, Oct. 1994.