Open Computing Grid for Molecular Science and Engineering

Sulev Sild,* Uko Maran, Andre Lomaka, and Mati Karelson

Department of Chemistry, University of Tartu, Tartu, Jakobi 2, 51014 Estonia

Received August 30, 2005

Grid is an emerging infrastructure for distributed computing that provides secure and scalable mechanisms for discovering and accessing remote software and data resources. Applications built on this infrastructure have great potential for addressing and solving large scale chemical, pharmaceutical, and material science problems. The article describes the concept behind grid computing and will present the OpenMolGRID system that is an open computing grid for molecular science and engineering. This system provides grid enabled components, such as a data warehouse for chemical data, software for building QSPR/QSAR models, and molecular engineering tools for generating compounds with predefined chemical properties or biological activities. The article also provides an overview about the availability of chemical applications in the grid.

INTRODUCTION

The rapid evolution of computer hardware has made available massive amounts of computing power at very affordable costs. Inexpensive personal computers (PCs) today have computational speed and storage capacities that equal or exceed a decade old supercomputer at the fraction of its cost. Despite the impressive hardware developments researchers still face different limitations when they try to use the full power of computers for solving complex scientific problems. To a large extent, the users are not hindered by the lack of hardware capabilities but by various limitations in the software applications that are currently in use. Considering that in the majority of cases the problem solving task involves the analysis of distributed and heterogeneous data sources and the application of different software tools, the limitations fall mainly into three categories. First, most of the available knowledge and tools are geographically dispersed over a large number of research institutions, both academic and industrial. Second, as a rule, the different tools are often incompatible with each other and may even require different hard- and software platforms. Third, the high complexity of the methodology and software involved requires highly skilled experts who are often working at different locations. Current software is not optimally designed to adequately handle this kind of requirements. However, these and many other relevant problems can currently be addressed by the grid computing.

GRID COMPUTING AND CHEMISTRY

Grid[1] is an emerging infrastructure for distributed computing that provides secure and scalable mechanisms for discovering and using remote software and data resources. The concept of grid computing is often described with an analogy to electric power grids, where users can consume computer resources as easily as plugging their electric devices into any wall outlet and getting electricity from collective power plants. In a similar way, the grid infrastructure or middleware allows for the building of virtual organizations where participants aggregate their available resources to a shared pool and use them on an as-needed basis. Grid will have a great impact both to industrial and academic users. Industry has interests because it provides new possibilities to develop and expose their services. But more importantly, it enables the scientific cooperation between geographically distributed groups and allows them to work together in ways that were previously impossible.

Although in reality the grid is pretty much a "work in progress", grid systems are developing at a very rapid pace, and several different options for grid middleware are available. Among the most mature and well-known grid systems are UNICORE, Globus, Condor, and commercial products like GridMP and Entropia. In the following subsections we will provide an overview of chemical applications (summarized in Table 1) that are currently available within different grid middleware systems.

Table 1. Chemical Applications in the Grid

middleware	software application	grid application framework	ref
Globus	DOCK	VLAB, Nimrod/G	6
	Gamess	Nimrod/G	10
	Autodock	WISDOM	11
	FLEXX	WISDOM	11
	Gaussian98	QC Grid	15
	WIEN2k	ASKALON, CoG	20
	NAMD	BioCoRe	24
GridMP	THINK	Screensaver Lifesaver project	29
	LigandFit	Screensaver Lifesaver project	29
Entropia	Autodock	AIDS@Home	34
Condor	MOPAC 2003	WWMM	36
UNICORE	CPMD		41
	Gaussian98	BioGRID	43
	Gamess	BioGRID	43
	Amber	BioGRID	43
	PDB database	BioGRID	43
	Entrez database	BioGRID	43
	MOLGEO	OpenMolGRID	46
	MOPAC 7	OpenMolGRID	46
	CODESSA Pro/MDC	OpenMolGRID	46
	CODESSA Pro/MDA	OpenMolGRID	46
	NTP database	OpenMolGRID	46,52
	ECOTOX database	OpenMolGRID	46,52

* Corresponding author e-mail: sulev.sild@ut.ee.

10.1021/ci050354f CCC: $33.50
Published on Web 01/06/2006

Globus. The Globus Toolkit[2] is a well established and popular open source software toolkit used for building grid systems and applications. It does not provide a complete solution framework for building a grid system but rather a set of standard building blocks and tools for the development of application grids. Therefore the learning curve for building a grid application is not straightforward, because developers have to select appropriate components for resource management, storage management, security, and information services. This issue has been addressed by higher level middleware systems that combine components from the Globus toolkit (and elsewhere) for a ready-to-use solution. Some available options are CoG,[3] gLite,[4] and GAT.[5]

A typical example of a grid based chemical application is the virtual laboratory (VLAB) project that aims to speed up molecular modeling for drug design by exploiting geographically distributed resources.[6] In this case, the Globus middleware together with the Nimrod/G toolkit[7] for distributed parametric modeling was used for the scheduling of protein−ligand docking jobs with DOCK[8] software. To screen large data sets, the authors of the VLAB have developed a distributed chemical database management system for the storage of ligands. This kind of computational problem is perfectly suitable for grid computing because each ligand can be processed on an independent processor (i.e. calculations are data parallel) and the Nimrod/G software can automate the distributed screening process by dividing available ligands into smaller subsets and scheduling them for execution at available remote sites. In a similar way, the Nimrod/G has been used for the parametrization of group difference potentials with Gamess[9] software.[10]

A data challenge effort called Wide In Silico Docking On Malaria (WISDOM)[11] is another large scale docking application in the grid. They use Autodock[12] and FLEXX[13] software for analyzing protein−ligand interactions. The gLite middleware is used to access resources provided by the EGEE project. The EGEE (Enabling Grids for E-sciencE) project is funded by the European Commission and brings together experts and computational resources from over 27 countries with the common aim of building on recent advances in grid technology and developing a service grid infrastructure which is available to scientists 24 h-a-day.[14]

A strong motivation for chemical applications in the grid is to provide better access to shared and distributed computational resources. For example, the Tsukuba Advanced Computer Center has developed a Quantum Chemistry Grid (QC Grid) for secure access to distributed (High Performance Computing) HPC resources.[15] A Web portal is provided as a user interface for the submission of ab initio calculation jobs for Gaussian98 software.[16] They have developed an intelligent meta-scheduler that is used for the optimal allocation of available resources at their HPC facilities for a large number of jobs. The input and output data for completed calculations are stored in a database and used for the estimation of computation time for subsequent jobs from the accumulated results to determine the most effective scenario of the resource usage.

A somewhat similar effort[17] to develop a Web portal for submitting ab initio calculations with Gamess[9] is performed under the National Science Foundation Middleware Initiative[18] project. Although, this effort is more focused on improving the usability of executing simple workflows consisting of common tasks in ab initio quantum chemical studies. For example, a common workflow handled by the system consists of performing a Gamess calculation, post-processing the results with PLTORB3D (from Gamess package), visualizing the results with QMView,[19] and, finally, storing results in a database.

ASKALON[20] is a grid application development and engineering environment that provides support for workflow and parametric study applications. It uses CoG for grid access over a Web portal and provides services for resource scheduling and monitoring, information services, and performance analysis and prediction. The ASKALON system has been tested for executing workflows with WIEN2k[21] software for the ab initio modeling of electronic structure of solids.[22]

Finally there are some applications for enabling a collaborative work by using the grid technology. A "knowledge grid" for informatics based combustion chemistry research has been developed by the Collaboratory for the Multi-scale Chemical Science (CMCS) project.[23] The collaboration and metadata based data management technologies are used to develop a community Web portal that integrates relevant chemistry resources and provides access to specific applications used in combustion chemistry. The CoG toolkit is used for accessing the grid services.

The Biological Collaborative Research Environment (BioCoRe)[24] software has been designed for collaborative work on computational structural biology. It has a Web interface for collaborative tools that can be used for recording research activities, preparing multiauthor documents, and organizing electronic conferences. The Web interface is also available for various simulation, analysis, and visualization software. Currently available applications include NAMD[25] and visualization tools VMD[26] and JMV.[27] BioCoRe uses Globus to run submitted NAMD jobs on remote resources, such as remote workstations or dedicated computation servers.

GridMP. The GridMP middleware by United Devices[28] offers an enterprise grid solution to aggregate unused computational resources from idle desktop PCs (or servers) available in organization or enterprise. Often this type of solutions is called desktop grids. This middleware is being used for drug design applications by several pharmaceutical companies, such Novartis and Johnson & Johnson. Besides commercial deployments, United Devices has provided their technology to several massive scientific research projects. Currently the largest scale chemical application in grid is the Screensaver Lifesaver project[29] run by professor W. Graham Richards at the University of Oxford. They use the idle computing power donated by owners of over 3 million desktop PCs across the world to screen 3.5 billion molecules for cancer-fighting potential. The screening for potential drugs is performed with pharmacophore matching (THINK[29]) and molecular docking (LigandFit[30]) software that is distributed to participants as a screensaver application, hence the name. Based on the same idea, similar projects have been set up for finding cures for anthrax and smallpox.[31]

Entropia. Entropia[32] is an analogous grid middleware system to United Devices. The applications of Entropia middleware are similar, and it has been used for various docking and quantum chemical ab initio applications.[33] The Scripps Research Institute uses Entropia middleware for the

FightAIDS@Home project, where Autodock[12] software is used for discovering new drugs for fighting against AIDS[34] by seeking HIV-1 protease inhibitors. Currently the project involves around 14 000 volunteers who donate their unused computing power.

Condor. Condor[35] is a specialized workload management system for computer-intensive jobs that can be used to build grid systems, including the interoperability with computational resources that are managed by Globus. The World Wide Molecular Matrix (WWMM)[36] uses Condor for submitting MOPAC[37] calculations to distributed computational resources. The WWMM is a distributed peer-to-peer system that acts as a repository for molecular structures and their properties. The repository currently includes ca. 250 000 molecules from the NCI database together with molecular properties that were calculated with MOPAC 2003.

UNICORE. UNICORE (Uniform Interface to Computer Resources) is an open source grid system.[38,39] It offers a fully integrated solution consisting of a powerful and easy to use Java based graphical user interface and server side components. The key concept behind the UNICORE system is an Abstract Job Object (AJO), which describes user requests by means of workflows, dependencies, and resource requirements in an abstract way. This approach makes it possible to harness heterogeneous computer resources, because the AJO is translated into a concrete job at the remote UNICORE site. The security infrastructure provides a single sign-on to an available UNICORE server by standard X.509 public key cryptography. The X.509 certificates are used for mutual authentication between UNICORE clients and servers and ensures that each part of the AJO is prepared by the authorized user. The integration of new applications is straightforward to the UNICORE grid system through the plugin mechanism. Currently, several applications have been integrated to UNICORE by developing appropriate plugins. For example, the Car-Parrinello Molecular Dynamics (CPMD) package[40] has a plugin for the job preparation and visualization simulations.[41]

UNICORE middleware has been used to establish a European computational grid between leading HPC centers—EUROGRID.[42] BioGRID[43] has adapted the EUROGRID infrastructure for using molecular biology and quantum chemistry related applications available at HPC facilities. UNICORE plugins have developed within this project for Gaussian98, Gamess, Amber, data visualization, and database access. The data access tools cover for crystallographic (PDB[44]) and sequence (Entrez[45]) databases.

OPENMOLGRID SYSTEM

Open Computing Grid for Molecular Science and Engineering (OpenMolGRID)[46] is a Grid enabled system for molecular design and engineering applications. The main motivation for the development of OpenMolGRID system was in silico testing, which has become a crucial part in the molecular design process of new drugs, pesticides, biopolymers, and biomaterials. In a typical design process, hundreds of thousands or even millions of candidate molecules are generated, and their viability has to be tested. Economically it is not feasible to carry out an experimental testing on all possible candidates. Therefore, computational screening methods are used to reduce the number of candidates to a more manageable size, as a cheaper and more time effective alternative. The OpenMolGRID system includes three main application components: data warehousing, data mining, and molecular engineering (see Figure 1). The data warehouse collects and handles molecular structures and associated data, the data mining part is used for building predictive QSPR/QSAR models, and the molecular engineering component is used for generating and selecting new molecular structures that match the given target property/activity values.

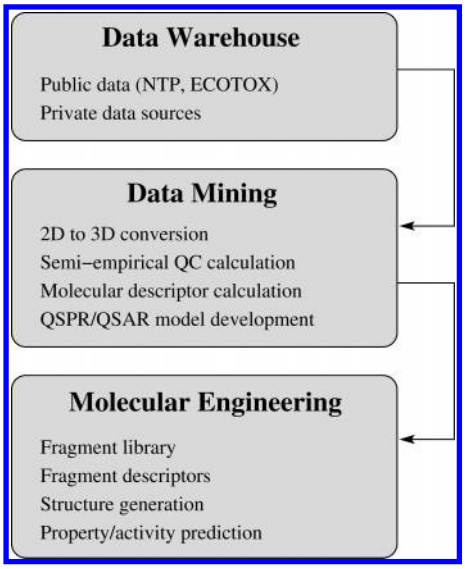

Figure 1. Main components of the OpenMolGRID system.

The OpenMolGRID system is based on the UNICORE middleware and uses the standard UNICORE plugin mechanism to integrate various software packages and data sources. The system is designed around a service oriented architecture where an abstraction layer, called the Abstract Resource Interface (ARI), is used to access software resources in a generic fashion. The ARI provides information to UNICORE clients with the resource description and defines standardized XML formats for input and output data. Data sources are integrated in the same general way using the ARI. At the same time, it extends the standard UNICORE client by adding new functionality in the area of workflow support and resource management.

The integration of existing and new software packages to the system is straightforward. The main idea behind the OpenMolGRID approach is to separate the in silico screening workflow into specific well-defined applications. Such applications include the generation of 3D coordinates for a molecule from its connectivity information, the calculation of molecular descriptors, the development of QSPR/QSAR models, etc. Each application uses UNICORE metadata facilities to advertise its functionality and input/output data types to UNICORE clients. The application metadata and application neutral data types are defined in XML language.[47] For example, the Chemical Markup Language (CML)[48] is used for the representation of molecular structures. This facilitates the design of generic application classes and transforms the actual software package to an abstract resource interface (see Figure 2). This approach allows for the reuse of the same client plugin for different applications within the same application class and makes it easier to exchange data between different applications in the system.

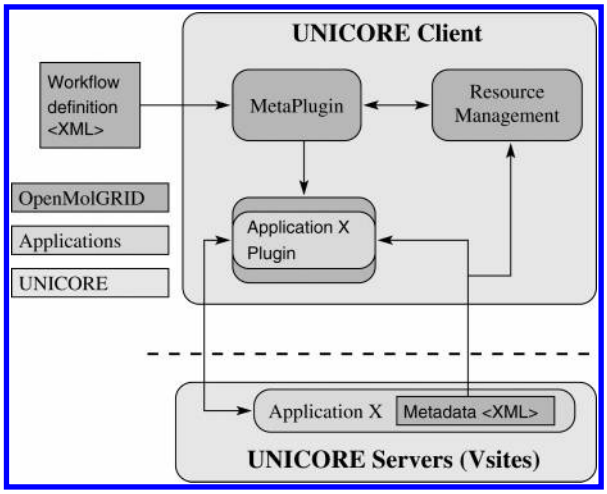

Figure 2. OpenMolGRID architecture for workflow support.

It is also necessary to develop a client component or plugin for the UNICORE client that is responsible for the definition of input data, the submission of a job, and finally for the visualization of a result for the given application. The client sends the submitted job object together with input data to the appropriate target system (*Vsite*) where it will be processed and executed. After the execution, the UNICORE client can fetch the outcome for visualization and storage. Although the client plugins are designed for users, they can be controlled by a workflow engine to automate various tasks.

The workflow engine, called MetaPlugin, supports the user in dealing with complex workflows. It builds UNICORE jobs from XML based workflow descriptions,[49,50] where the UNICORE system specific details such as explicit file transfers, task dependencies, and resource allocation are taken care of automatically. Manual data conversions are eliminated by automatically adding suitable data transformation tasks between subsequent tasks that have different output and input formats. Computationally intensive tasks can be run on multiple sites, if the input data can be split into smaller pieces and distributed. The resources needed for the job execution are identified and allocated automatically by the MetaPlugin.

Data Warehouse. Predictive QSPR/QSAR modeling requires the handling and management of experimentally measured property or activity data together with computationally derived data (e.g. molecular descriptors) that describe the relevant chemical structures. The experimental data are often not readily available but must be retrieved from proprietary or public data repositories. Furthermore, the data must be integrated and formatted to be amenable to data mining methods such as multiple regression analysis, partial least squares, or artificial neural networks. In the next step, theoretically derived data must be calculated. Frequently different data sets overlap partially, and some data are therefore recalculated needlessly many times. Data warehousing[51] is thus often employed to provide the data integration and formatting functionality needed by data mining applications. A data warehouse integrates, cleanses, normalizes, and consolidates data from different sources and transforms them onto "ready-to-use" data structures.

The data warehouse component[52] in the OpenMolGRID system provides a grid-enabled environment that harvests geographically dispersed data sources across the Internet and stores transformed data for data mining in a repository accessible to the OpenMolGRID users. The experimental property/activity data are normalized in order to make data mining more convenient. This process includes the removal of inconsistent entries and the standardization of data based on requirements (e.g. conversion units, the calculation of inverse logarithms from toxicity measurements, etc.). The data warehouse makes experimental data available together with the molecular descriptor values. Since most public data sources do not include molecular descriptors, those will be computed on demand. Grid resources are used in this process as a significant amount of computational resources are required. The data access to the data warehouse is provided with a generic data access tool and the respective client plugin that can be used in workflows to pull relevant data for QSPR/QSAR modeling and molecular engineering tools.

QSPR/QSAR Modeling. The QSPR/QSAR models are developed by finding mathematical relationships between experimentally measured property/activity and molecular structures encoded through the respective descriptors. The development of models usually involves a multistep workflow where certain tasks are carried out at the predetermined order. A typical workflow starts with the selection of a training set data with the experimental property/activity values from the data source (e.g. database, file system). Thereafter, the molecular descriptors are calculated to represent molecular structures in the training set. The descriptor calculation itself can be a multistep process and includes the generation of three-dimensional atomic coordinates and carrying out quantum-chemical calculations of individual molecules. Finally QSPR/QSAR models are developed with mining techniques (e.g. MLR, PLS, ANN, etc.) and validated to make sure that significant descriptors selected for the statistical model have a causal relationship with the modeled property or activity. The OpenMolGRID system provides grid interfaces for performing the above-mentioned tasks individually or in combination in complex workflows. The following applications are available for the QSPR/QSAR modeling.

3D Structure Generation. Fast 2D to 3D conversion methods are needed to generate random or partly optimized conformations necessary for 3D descriptor calculation and quantum chemical calculations. Most of the 2D to 3D conversion programs generate 3D structures by assigning some average values to bond lengths and valence angles. The most commonly used method is the distance geometry approach developed by Crippen et al.[53] In OpenMolGRID, the MOLGEO program[54] has been adapted that implements distance geometry and recursive depth first search (DFS) algorithms. This step is omitted when 3D and quantum chemical descriptors are not needed.

Semiempirical Calculations. The calculation of quantum chemical descriptors[55] requires a preliminary step of the respective molecular wave function calculation. Typically, semiempirical methods (with parametrizations such as AM1[56] and PM3[57]) are preferred over ab initio methods because of their lower computational cost. Despite that, quantum chemical calculations are frequently still the most computationally demanding step in the QSAR/QSPR workflow. The computational cost of most quantum chemical methods depends polynomially on the size of the molecule that makes them especially time-consuming for large-size

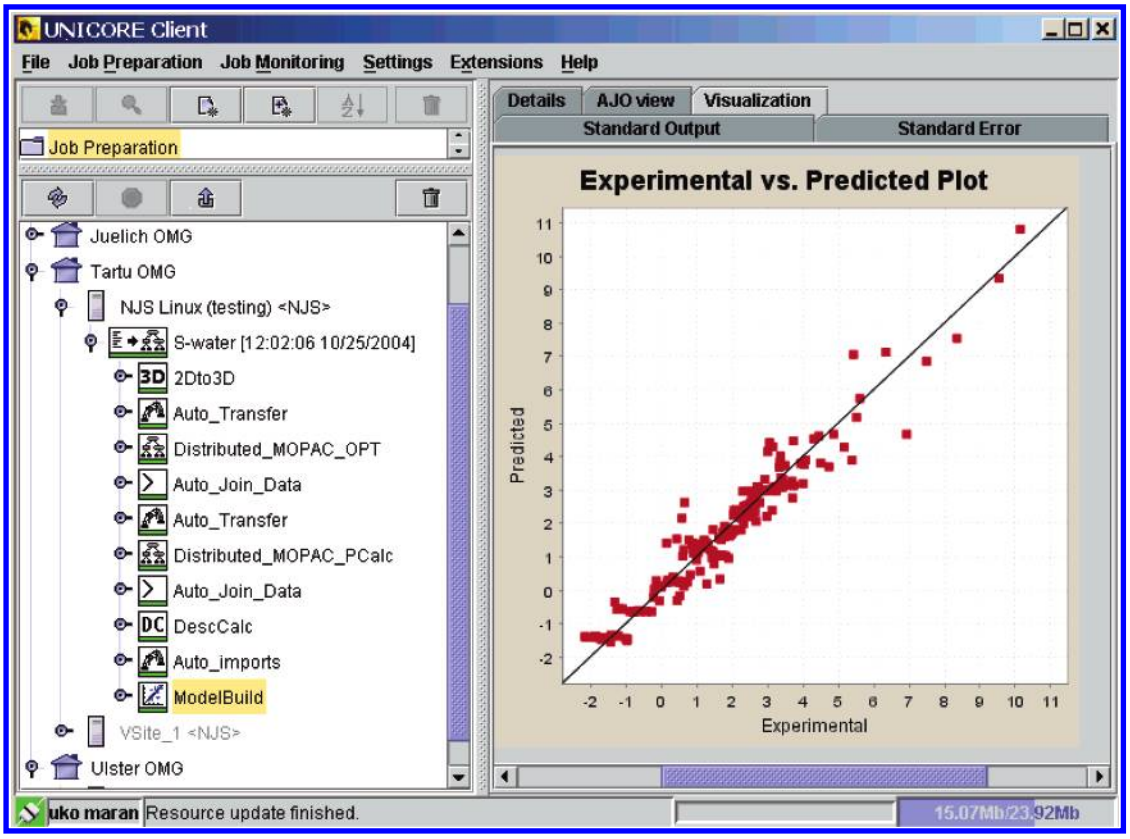

Figure 3. Job monitoring area with finished workflow for the QSPR model building.

molecules. OpenMolGRID uses a modified version of MOPAC (version 7)[37] to generate input data for quantum chemical descriptor calculation. MOPAC is a general-purpose semiempirical quantum mechanics package for the study of chemical properties and reactions in gas, solution, or solid state.

Descriptor Calculation. The enormous number of descriptors available today can be generated by various software packages. Some packages limit themselves to a certain class of descriptors, whereas others generate a selection of several of them. CODESSA Pro's molecular descriptor calculation module (MDC) that has been integrated with OpenMolGRID enables the calculation of a large variety of constitutional, topological, geometrical, electrostatic, quantum-chemical, and thermodynamical descriptors.[58] Typically, MDC allows the calculation of approximately 600 descriptors for each compound. Depending on the types of descriptors needed, MDC can use either 2D or 3D structures with optional MOPAC calculation results as input.

Model Building. Several mathematical approaches can be used to find the functional relationship between the molecular property and the descriptors. Most common is the simple multilinear regression on the appropriate descriptor set. An important part of model development is the so-called descriptor selection, where the most relevant subset of descriptors is selected for the particular QSAR/QSPR model. Because of the large number of descriptor combinations that can be selected from the available descriptor pool, the computational cost of model building increases rapidly with the size of the descriptor pool. Several software packages exist that allow the generation of QSAR/QSPR models by various statistical or neural net methods, calculate statistical parameters, validate the models, etc. OpenMolGRID has integrated the molecular descriptor analysis (MDA) module from the CODESSA Pro software for the QSAR/QSPR model development.[58] MDA allows for the performance of multilinear regression or partial least-squares analysis, calculates various statistical parameters for the models, studies the intercorrelation of the different descriptors, and validates the models by randomization test and leave-one-out and leave-many-out cross-validation. Several descriptor selection algorithms are available for the search of the most optimal subset of descriptors.

Molecular Engineering. Molecular engineering is the task of designing molecular compounds and materials with predetermined target chemical properties or biological activities. It is an important activity in the development of many products, including chemicals, materials, detergents, and drugs. The traditional approach is to synthesize and test the set of molecules experimentally for their properties. This is both time-consuming and expensive, which considerably delays the arrival of new products to the market. Computational methods give the unique chance to accelerate this process by prescanning an available or generated set of candidate molecules to reduce the number of potentially active molecules to be synthesized and tested experimentally.

The molecular engineering environment in the OpenMolGRID system provides several tools that make it possible to carry out molecular engineering tasks in the Grid environment. It includes a fragment library, structure generation application, prediction, and filtering tools. The structure generator uses fragment structures for the construction of a large number of molecular structures. The structure generation algorithm can optionally use fragment descriptors for reducing the amount of generated structures. The properties and activities of generated candidate structures are validated

against the target values with relevant QSPR/QSAR models, and, finally, a small subset of molecules is selected that are best matching the targeted properties. This process can be automated by workflows, and the compute intensive filtering process can be sped up by distributing computations over multiple UNICORE sites.

Examples. The OpenMolGRID system has been tested for various applications. The data warehousing component has been used to make toxicity data available for the data mining environment from selected public data sources.[52] Currently two data resources have been integrated. The first is the National Toxicology Program (NTP) database,[59] which provides information about potentially toxic chemicals for health regulatory and research agencies. The second is the ecotoxicology (ECOTOX) database,[60] which provides chemical toxicity information for aquatic and terrestrial life.

The automated workflow support has been successfully tested for building QSPR/QSAR models. A recent test for building a QSPR model for solubility of organic compounds in water showed that the grid based approach significantly speeds up the model development process[49] by eliminating manual interaction between subtasks and distributing semiempirical quantum chemical calculations over multiple target sites. The outcome of the finished workflow is depicted in Figure 3. The left side of the figure shows subtasks executed in the model building workflow. In the original workflow description five subtasks were specified. All subtask names prefixed with "Auto_" were added by the workflow engine (MetaPlugin) to execute the workflow with grid resources that were available at the submission time. The right side of the figure is used for displaying the outcome of the selected subtask. In this figure it shows the visualization panel with experimental vs predicted plot for the water solubility model. The system has also been used for the modeling of cytotoxicity on a proprietary data set of 30 000 compounds in order to discover new potential anticancer agents.

SUMMARY

We have provided an overview (see Table 1 for a summary) on the availability of chemical applications in grid and described the architecture and functionality of the OpenMolGRID system. Currently many different applications exist in the grid that use different available middleware options. Undoubtedly more chemical and biochemical grid applications will be developed. Also, alternative grid middleware systems may emerge in the future. Considering that all major grid systems are currently moving toward a more generic grid services based architecture, the interoperability between applications developed under different middleware systems should be greatly improved in the future. Facilitating thus scientific problem solving in a more powerful and innovative environment.

Our experience with the development of the OpenMolGRID system shows that the grid based approach by using the UNICORE middleware is very well suited for a wide array of chemical applications and particularly well for molecular design and engineering applications. The main strength of the UNICORE middleware is its plugin mechanism that allows easy integration of legacy applications to the grid. We have used a service oriented approach for integrated applications that makes it easy to combine them in complex workflows and will make it easier to keep up with future developments of the grid middleware systems. The UNICORE middleware is still evolving, it has active user and developer community, and with each release new features are added and its scalability is improved.

One of the biggest assets in the OpenMolGRID system is the workflow engine that facilitates the execution of complex workflows in a distributed problem solving environment, where multiple scientific applications can be combined to solve different scientific problems. We have used it successfully for building QSPR/QSAR models, semiempirical quantum chemical calculations, constructing molecular structures from fragment structures, and screening these structures with predictive models. In addition to applications, it integrates and provides access mechanisms to heterogeneous and globally distributed data sources. Currently it is being used and tested in different academic institutions and pharmaceutical companies in Europe.

ACKNOWLEDGMENT

This research was funded by the European Union under the Fifth Framework of the Information Society Technologies Program as part of the OpenMolGRID project (IST-2001-37238). Many colleagues have contributed to the OpenMolGRID project. We thank them all and, in particular, M. Romberg, Dr. B. Schuller, Dr. E. Benfenati, Dr. Mose Casalegno, Dr. P. Mazzatorta, Dr. F. Darvas, A. Papp, I. Bagyi, Dr. W. Dubitzky, D. McCourt, and Dr. G. H. Diercksen.

REFERENCES AND NOTES

(1) Foster, I.; Kesselman, C. *The Grid 2: Blueprint for a New Computing Infrastructure*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, 2003.

(2) Globus Toolkit Web site: http://www.globus.org/.

(3) Commodity Grid Kits, University of Chicago, http://www.globus.org/cog/.

(4) Lightweight Middleware for Grid Computing (gLite) Web site: http://cern.ch/glite/.

(5) Allen, G.; Davis, K.; Goodale, T.; Hutanu, A.; Kaiser, H.; Kielmann, T.; Merzky, A.; van Niewpoort, R.; Reinefeld, A.; Schintke, F.; Schütt, T.; Seidel, E.; Ullmer, B. The Grid Application Toolkit: Toward Generic and Easy Application Programming Interfaces for the Grid. In *Proceedings of the IEEE*; IEEE: 2005; Vol. 93, pp 534−550.

(6) Buyya, R.; Branson, K.; Giddy, J.; Abramson, D. The Virtual Laboratory: a Toolset to Enable Distributed Molecular Modelling for Drug Design on the World-Wide Grid. *Concurrency and Computation: Practice and Experience* **2003**, *15*, 1−25.

(7) Buyya, R.; Abramson, D.; Giddy, J. Nimrod-G: an Architecture for a Resource Management and Scheduling System in a Global Computational Grid. In *Proceedings of the HPC ASIA'2000, China*; IEEE CS Press: U.S.A., 2000.

(8) Shoichet, B. K.; Bodian, D. L.; Kuntz, I. D. Molecular Docking Using Shape Descriptors. *J. Comput. Chem.* **1992**, *13*, 380−397.

(9) Schmidt, M. W.; Baldridge, K. K.; Boatz, J. A.; Elbert, S. T.; Gordon, M. S.; Jensen, J. H.; Koseki, S.; Matsunaga, N.; Nguyen, K. A.; Su, S. J.; Windus, T. L.; Dupuis, M.; Montgomery, J. A. General Atomic and Molecular Electronic Structure System. *J. Comput. Chem.* **1993**, *14*, 1347−1363.

(10) Sudholt, W.; Baldridge, K. K.; Abramson, D.; Enticott, C.; Garic, S.; Kondric, C.; Nguyen, D. Application of Grid Computing to Parameter Sweeps and Optimizations in Molecular Modeling. *Future Gen. Comput. Syst.* **2005**, *21*, 27−35.

(11) Wide In Silico Docking On Malaria (WISDOM) Web site: http://wisdom.eu-egee.fr/.

(12) Morris, G. M.; Goodsell, D. S.; Halliday, R. S.; Huey, R.; Hart, W. E.; Belew, R. K.; Olson, A. J. Automated Docking Using a Lamarckian Genetic Algorithm and Empirical Binding Free Energy Function. *J. Comput. Chem.* **1998**, *19*, 1639−1662.

(13) Rarey, M.; Kramer, B.; Lengauer, T.; Klebe, G. A Fast Flexible Docking Method using an Incremental Construction Algorithm. *J. Mol. Biol.* **1996**, *261*, 470−489.

(14) Enabling Grids for E-sciencE (EGEE) Web site: http://public.eu-egee.org/.

(15) Nishikawa, T.; Nagashima, U.; Sekiguchi, S. Design and Implementation of Intelligent Scheduler for Gaussian Portal on Quantum Chemistry Grid. In *International Conference on Computational Science − ICCS 2003*; Sloot, P. M. A., Abramson, D., Bogdanov, A. V., Dongarra, J., Zomaya, A. Y., Gorbachev, Y. E., Eds.; Lecture Notes in Computer Science, Springer-Verlag: Berlin, Heidelberg, 2003; Vol. 2659, pp 244−253.

(16) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Montgomery, J. A., Jr.; Vreven, T.; Kudin, K. N.; Burant, J. C.; Millam, J. M.; Iyengar, S. S.; Tomasi, J.; Barone, V.; Mennucci, B.; Cossi, M.; Scalmani, G.; Rega, N.; Petersson, G. A.; Nakatsuji, H.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Klene, M.; Li, X.; Knox, J. E.; Hratchian, H. P.; Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.; Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Ayala, P. Y.; Morokuma, K.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.; Zakrzewski, V. G.; Dapprich, S.; Daniels, A. D.; Strain, M. C.; Farkas, O.; Malick, D. K.; Rabuck, A. D.; Raghavachari, K.; Foresman, J. B.; Ortiz, J. V.; Cui, Q.; Baboul, A. G.; Clifford, S.; Cioslowski, J.; Stefanov, B. B.; Liu, G.; Liashenko, A.; Piskorz, P.; Komaromi, I.; Martin, R. L.; Fox, D. J.; Keith, T.; Al-Laham, M. A.; Peng, C. Y.; Nanayakkara, A.; Challacombe, M.; Gill, P. M. W.; Johnson, B.; Chen, W.; Wong, M. W.; Gonzalez, C.; Pople, J. A. *Gaussian 98*; Gaussian, Inc.: Wallingford, CT, 1998.

(17) Greenberg, J. P.; Mock, S.; Bhatia, K.; Katz, M.; Bruno, G.; Sacerdoti, F.; Papadopoulos, P.; Baldridge, K. K. Incorporation of Middleware and Grid Technologies to Enhance Usability in Computational Chemistry Applications. *Future Gen. Comput. Syst.* **2005**, *21*, 3−10.

(18) National Science Foundation Middleware Initiative at http://www.nsf-middleware.org/default.aspx.

(19) Baldridge, K.; Greenberg, J. QMView: a Computational 3D Visualization Tool at the Interface between Molecules and Mankind. *J. Mol. Graph.* **1995**, *13*, 63−66.

(20) ASKALON Programming Environment for Grid Computing Web site: http://http://dps.uibk.ac.at/projects/askalon/.

(21) Blaha, P.; Schwarz, K.; Madsen, G. K. H.; Kvasnicka, D.; Luitz, J. WIEN2k, An Augmented Plane Wave + Local Orbitals Program for Calculating Crystal Properties; Kalheinz Schwartz, Technical University of Wien, Austria, ISBN 3-9501031-1-2, 2001; http://www.wien2k.at/.

(22) Wieczorek, M.; Prodan, R.; Fahringer, T. Scheduling of Scientific Workflows in the ASKALON Grid Environment. *ACM SIGMOD Rec.* **2005**, *34*, 56−62.

(23) Myers, J. D.; Allison, T. C.; Bittner, S.; Didier, B. T.; Frenklach, M.; Green, W. H.; Ho, Y.-L.; Hewson, J. C.; Koegler, W. S.; Lansing, C.; Leahy, D.; Lee, M.; McCoy, R.; Minkoff, M.; Nijsure, S.; von Laszewski, G.; Montoya, D.; Pancerella, C. M.; Pinzon, R.; Pitz, W.; Rahn, L. A.; Ruscic, B.; Schuchardt, K.; Stephan, E. G.; Wagner, A.; Windus, T. L.; Yang, C. L. A Collaborative Informatics Infrastructure for Multi-scale Science. In *2nd International Workshop on Challenges of Large Applications in Distributed Environments (CLADE 2004)*; IEEE Computer Society: 2004; pp 24−33.

(24) Bhandarkar, M.; Budescu, G.; Humphrey, W. F.; Izaguirre, J. A.; Izrailev, S.; Kale, L. V.; Kosztin, D.; Molnar, F.; Phillips, J. C.; Schulten, K. BioCoRE: A Collaboratory for Structural Biology. In *Proceedings of the SCS International Conference on Web-Based Modeling and Simulation*; Bruzzone, A. G., Uchrmacher, A., Page, E. H., Eds.; Society for Computer Simulation: San Francisco, CA, 1999; pp 242−251.

(25) Kalé, L.; Skeel, R.; Bhandarkar, M.; Brunner, R.; Gursoy, A.; Krawetz, N.; Phillips, J.; Shinozaki, A.; Varadarajan, K.; Schulten, K. NAMD2: Greater Scalability for Parallel Molecular Dynamics. *J. Comput. Phys.* **1999**, *151*, 283−312.

(26) Humphrey, W.; Dalke, A.; Schulten, K. VMD: Visual Molecular Dynamics. *J. Mol. Graph.* **1996**, *14*, 33−38.

(27) JMV is developed by the Theoretical and Computational Biophysics Group in the Beckman Institute for Advanced Science and Technology at the University of Illinois at Urbana-Champaign with NIH support, Web page: http://www.ks.uiuc.edu/Research/jmv/.

(28) United Devices Web site: http://www.ud.com/.

(29) Richards, W. G. Virtual Screening Using Grid Computing: The Screensaver Project. *Nature Rev. Drug Discovery* **2002**, *1*, 551−555.

(30) Venkatachalam, C. M.; Jiang, X.; Oldfield, T.; Waldman, M. LigandFit: a Novel Method for the Shape-directed Rapid Docking of Ligands to Protein Active Sites. *J. Mol. Graphics Modell.* **2003**, *21*, 289−307.

(31) Richards, W. G.; Grant, G. H.; Harrison, K. N. Combating Bioterrorism with Personal Computers. *J. Mol. Graphics Modell.* **2004**, *22*, 473−478.

(32) Entropia at http://www.entropia.com/.

(33) Chien, A.; Calder, B.; Elbert, S.; Bhatia, K. Entropia: Architecture and Performance of an Enterprise Desktop Grid System. *J. Parallel Distrib. Comput.* **2003**, *63*, 597−610.

(34) FightAIDS@Home Web site: http://fightaidsathome.scripps.edu/.

(35) Thain, D.; Tannenbaum, T.; Livny, M. Distributed Computing in Practice: the Condor Experience. *Concurrency-Pract. Ex.* **2005**, *17*, 323−356.

(36) World Wide Molecular Matrix, http://wwmm.ch.cam.ac.uk/wwmm.html.

(37) Stewart, J. J. P. MOPAC: a Semiempirical Molecular Orbital Program. *J. Comput.-Aided Mol. Des.* **1990**, *4*, 1−45.

(38) UNICORE Web site: http://unicore.sourceforge.net/.

(39) Romberg, M. The UNICORE Grid Infrastructure. *Scientific Programming* **2002**, *10*, 149−157.

(40) Marx, D.; Hutter, J. Ab Initio Molecular Dynamics: Theory and Implementation. In *Modern Methods and Algorithms of Quantum Chemistry*; NIC Series: John von Neumann Institute for Computing, Jülich, 2000; Vol. 3, pp 329−477.

(41) Huber, V. Supporting Car-Parrinello Molecular Dynamics with UNICORE. In *International Conference on Computational Science − ICCS 2001, Pt. 1*; Springer-Verlag: 2001; Vol. 2073, pp 560−566.

(42) Lesyng, B.; Bala, P.; Erwin, D. EUROGRID: European Computational Grid Testbed. *J. Parallel Distrib. Comput.* **2003**, *63*, 590−596.

(43) Pytlinski, J.; Skorwider, L.; Bala, P.; Nazaruk, M.; Wawruch, K. BioGRID − Uniform Platform for Biomolecular Applications. In *Euro-Par '02: Proceedings of the 8th International Euro-Par Conference on Parallel Processing*; Monien, B., Feldman, R., Eds.; Lecture Notes in Computer Science, Springer-Verlag: London, U.K., 2002; Vol. 2400, pp 881−884.

(44) Berman, H. M.; Westbrook, J. D.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. The Protein Data Bank. *Nucleic Acids Res.* **2000**, *28*, 235−242.

(45) Schuler, G. D.; Epstein, J.; Ohkawa, H.; Kans, J. A. Entrez: Molecular Biology Database and Retrieval System. *Methods Enzymol.* **1996**, *266*, 141−161.

(46) OpenMolGRID Web site: http://www.openmolgrid.org/.

(47) XML Specification: http://www.w3.org/TR/2004/REC-xml11-20040204/.

(48) Murray-Rust, P.; Rzepa, H. S. Chemical Markup Language and XML. 1. Basic principles. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 928−942.

(49) Sild, S.; Maran, U.; Romberg, M.; Schuller, B.; Benfenati, E. OpenMolGRID: Using Automated Workflows in GRID Computing Environment. In *Advances in Grid Computing − EGC 2005, European Grid Conference*; Sloot, P. M. A., Hoekstra, A. G., Priol, T., Reinefeld, A., Bubak, M., Eds.; Lecture Notes in Computer Science, Springer: Germany, 2005; Vol. 3470, pp 464−473.

(50) Schuller, B.; Romberg, M.; Kirtchakova, L. Application Driven Grid Developments in the OpenMolGRID Project. In *Advances in Grid Computing − EGC 2005, European Grid Conference*; Sloot, P. M. A., Hoekstra, A. G., Priol, T., Reinefeld, A., Bubak, M., Eds.; Lecture Notes in Computer Science, Springer: Germany, 2005; Vol. 3470, pp 23−29.

(51) Moss, L.; Adelman, A. Data Warehousing Methodology. *J. Data Warehousing* **2000**, *5*, 23−31.

(52) Dubitzky, W.; McCourt, D.; Galushka, M.; Romberg, M.; Schuller, B. Grid-enabled Data Warehousing for Molecular Engineering. *Parallel Comput.* **2004**, *30*, 1019−1035.

(53) Crippen, G. M. *Distance Geometry and Conformational Calculations*; Research Studies: New York, 1981.

(54) Gordeeva, E.; Katritzky, A. R.; Shcherbukhin, V. V.; Zefirov, N. S. Rapid Conversion of Molecular Graphs to Three-dimensional Representation Using the MOLGEO Program. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 102−111.

(55) Karelson, M.; Lobanov, V. S.; Katritzky, A. R. Quantum-Chemical Descriptors in QSAR/QSPR Studies. *Chem. Rev.* **1996**, *96*, 1027−1043.

(56) Dewar, M. J. S.; Zoebisch, E. G.; Healy, E. F.; Stewart, J. J. P. Development and Use of Quantum Mechanical Molecular Models. 76. AM1: A New General Purpose Quantum Mechanical Molecular Model. *J. Am. Chem. Soc.* **1985**, *107*, 3902−3909.

(57) Stewart, J. J. P. Optimization of Parameters for Semiempirical Methods. 1. Method. *J. Comput. Chem.* **1989**, *10*, 209−220.

(58) CODESSA PRO Web site: http://www.codessa-pro.com/.

(59) National Toxicology Program server at http://ntp-server.niehs.nih.gov/.

(60) ECOTOX (ECOTOXicology) database at http://www.epa.gov/ecotox/.

CI050354F