TOWARDS UTILISATION OF THE GRID IN ADVANCED ENVIRONMENTAL DECISION SUPPORT

Michael J. Mineter[1], Andreas N. Skouloudis[1], Claire Jarvis[2] and Steve Dowers[2]
[1]Institute for Environment and Sustainability, Joint Research Centre Ispra,
European Commission, TP. 280, Ispra (VA), I-21020, Italy
[2]Department of Geography, University of Edinburgh, Drummond St,
Edinburgh EH8 9XP, Scotland

INTRODUCTION

This paper explores the impact that new Internet-based computing paradigms will make upon environmental decision support systems (DSS). Emerging Grid technologies will enable powerful and flexible DSS that integrate models, GIS, data collection, archiving and data mining with resources in computation and data storage. To illustrate the possibilities, air quality modelling is considered, focussing upon the development of DSS intended to reduce the magnitude and frequency of major pollution events in the urban environment. The move to Grid-based DSS will accentuate issues already critical in changing the use of a model from research to DSS mode.

The current period in environmental modelling, and in particular air quality modelling is one of great potential as many models have reached a state in which they are sufficiently reliable to be integrated in environmental management systems. This maturity has coincided with revolutionary changes in internet technologies, in both hardware and software. The first major step was to support easy *document*-retrieval (the Web). The second, underway now, is to support *data* generation and retrieval using Web services, where these are run on hardware remote from the user, managed by a service provider. The third step will be to Grid computing: flexible, distributed computing, in which resources on the Internet are perceived as a "*virtual computer*" enabled by internet bandwidths of many Gigabits per second. In effect the internet becomes the backbone of a distributed computer for the duration of a particular application.

Grid is the enabling technology for eScience, whereby geographically distributed "virtual organisations" of people collaborate in undertaking science by sharing and dynamically configuring geographically distributed resources (for example, data generators such as air quality instruments, data archives, computers, network bandwidth and software) into virtual computers, connected by the internet (Foster et al., 2001). The publication of the first book on Grid (Foster and Kesselman, 1999) is seen as a milestone in the establishment of the key concepts. The term Grid is a metaphor arising from the ease with which electricity is used, with no regard to its origin (the generator). These eScience methods are also directly applicable to the world of environmental policy making and decision support – the virtual organisations might include policy makers, environmental managers, environmental scientists, GIS specialists, data providers, meteorologists, economists, geographers, amongst others.

The term "middleware" is used for the software that will enable the Grid to function, by addressing fundamental issues including resource management, security, service creation and closure. Supported by meetings of the Global Grid Forum (2002), and driven by very strong commercial and academic initiatives, the "Open Grid Services Architecture" (Foster et al., 2002) is giving shape to middleware and also to application software that will be built upon it. OGSA describes middleware in terms of services, defining their interfaces in terms of messages (thus making no statement about implementation).

OGSA uses a superset of the languages emerging from Web services (W3C, 2002) primarily XML in various guises for specific purposes, to maximise the overlap between Web and Grid services. In the Grid environment when a service is made available its metadata and a definition of its interfaces are stored in a registry. When a user's software recognises the need to find the service, it interrogates the registry, receives metadata about each available service, selects the service of choice (and in future, the computational architecture required, based on issues such as desired run –time and available budget for its execution), creates a new instance of the service, and uses it. Each service can in turn invoke further services – so a requirement to "model the air quality of central Milan" would become an invocation of one service, and that in turn might invoke multiple further services comprising different aspects of the necessary modelling and perhaps data mining.

FROM MODELS TO DECISION SUPPORT

The effort entailed in moving a model from research to DSS use depends strongly upon the methods used to develop the initial code. Three aspects are emphasised: ease of development, modularity and support for parallel processing.

Models built for research purposes in general require re-engineering before use in decision support. This pattern recurs due to model-developers being time-constrained environmental scientists in some cases having minimal software design experience, but more importantly having sharply focused goals and inadequate application development environments. Software is thus developed to prove concepts, not to provide easy configuration for the wider demands of decision support, nor to allow easy integration with additional tools such as GIS (for display or for further analysis with contextual geographical data), nor for reusability, maintainability or easy user-interfaces. The need is for the initial development to be made simpler, and for that simplicity to be achieved in a development environment that imposes structure and modularity, and that secures a longer-term future for the code: routine issues like input and output should not be matters to be coded from scratch by the developer.

Amongst the application development environments leading to this panacea are commercial GIS packages, iconic modelling environments and framework-component architectures. GI Systems support storage, analysis and display of geographical data, but in some cases their datamodels can be extendable for specific model domains – leading to tightly coupled model and GIS applications, for example, in water resource management, (Maidment, 2002). Iconic systems are already established for some environmental models by Muetzelfeldt and Taylor (1997) and also Costanza and Gottlieb (1998). Prominent amongst framework-component approaches is Cactus (2002) in which a central framework invokes computational "thorns". Developed for, and primarily utilised in physics, to our knowledge it has yet to be used in environmental processing. Whilst both GIS and iconic environments can provide for easy and modular code development, the potential need for parallel processing is not yet supported. Cactus is designed for this, but the extent of its applicability to environmental problems is unproven – it illustrates facets of the solutions needed, at least.

Parallelisation is necessary if high performance is needed through utilising multiple processors on one task. For decision-support use, as compared to research use, additional computation may be necessary both because approximations valid for research may be too crude for decision support, and moreover because decision support with deterministic models should entail parameter sweeps to assess the sensitivity of results to changes in inputs and parameters. For research purposes, a hypothesis might be tested on a relatively small dataset; for DSS purposes larger datasets are often required. Research runs can be scheduled for overnight or weekend;

DSS requirements may be more urgent. Parallelisation is thus a major issue, even for codes run sequentially when in research mode.

In writing code for parallel processors, a range of generic tools that perform basic tasks of I/O, of scattering and gathering data from the multiple processors, and of synchronising these processors, are available. Mineter et al. (2000) discuss this for geographical problems, for which specific tools remain scarce. At present, for example, linking a parallelised model to a GIS would typically entail loose-coupling between the sequential GIS and the model, via the exchange of data files, without the benefits of the recent open interfaces to GIS data models.

Thus there remains the need to create easy-to use environments for application development, ones that support both data parallelism (processing of different subsets of data in similar ways, concurrently) and functional parallelism (with different sub-tasks – for example different objects' methods – being executed concurrently). Due to the scope for users of real-time applications, the service-orientation of the Grid is likely to accentuate the need for high performance modelling within one service, as well as support for concurrent execution of multiple services.

URBAN AIR QUALITY MODELLING IN THE NEW WORLD OF THE GRID

The age that is coming is where a manager with responsibility for urban pollution receives alerts that levels are forecast to rise in the next few days. Logging onto a Grid portal, the manager specifies "what-if" scenarios. Guided by expertise within the software, the manager determines the parameter sweeps of inputs needed, and the range of outputs required. The manager indicates the urgency with which results are required and negotiates a cost/performance contract with the software. The analyses are run on remote resources – the IT budget covers the time and resources that are used; there is a minimal internal IT department. The analyses use data-mining from previous incidents and a variety of model result archives, supplemented by runs with models newly made available by collaborating research groups, and at the agreed time an alert indicates that data are now available for further interactive analysis to support the necessary decisions.

Elements of the above story have been described and developed over several years, in seeking to bridge the gap from research use of models to policy decision support. Fedra (1999) recognised the viability of models was sufficient to meet the need to "embed air quality models in a conceptual framework that includes and explicitly addresses policy relevant elements such as the control of emission sources including economic criteria, monitoring of ambient air quality and the compliance with standards, and impacts on human health and the environment". He proposed a range of models that would be required "ranging from simple steady-state screening models to dynamic, 3D photochemical models which are implemented on parallel computers for better than real-time performance" and recognised that models could be implemented locally or on a remote compute server. Amongst on-going projects, SUTRA (Sustainable Urban Transportation) (SUTRA, 2002) includes work packages to seek policy-level decision support, integrating energy, traffic, and air pollution models with additional analytical components such as GIS. The Healthier Environment through Abatement of Vehicle Emissions and Noise (HEAVEN, 2002) project has developed an integrated Decision Support System (DSS) for urban traffic management.

The transition to Grid-based DSS will primarily be a matter of defining services at the right level of software (for example, a service might comprise a complete air quality model, or else be formed from modules within it). Each service will comprise legacy code wrapped in a "Grid-aware" code that manages the messages and defines the interfaces used by other services including middleware. Figure 1 gives a schematic example of Grid services for air quality modelling. In the Grid context, the components are services running on different computers, their interfaces defined by messages; access to data is provided by Grid-enabled services.

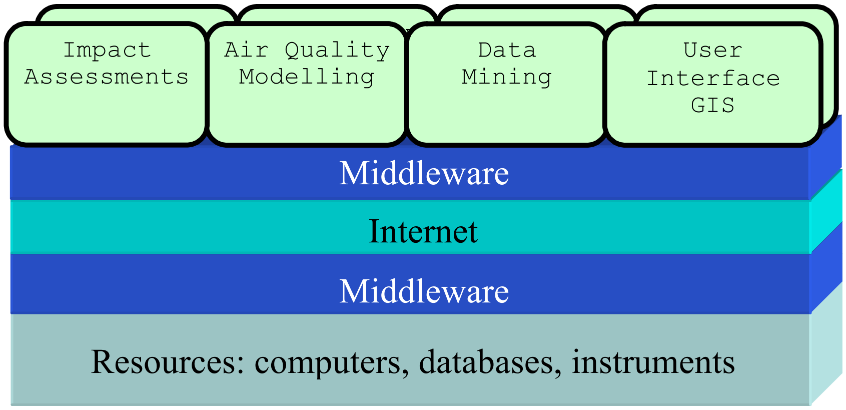

Figure 1 A schematic of air quality services built on Grid middleware.

The Grid is clearly not necessary for every DSS implementation – a web service or a locally delivered, installed, updated and managed program is also an alternative. However where large datasets, and distributed communities of users and distributed resources need to be coordinated, in particular for perhaps intermittent intense applications, the Grid will open new horizons. Grid technology is already available for parameter sweeps, under the San Diego Supercomputing Centre project, AppLeS Parameter Sweep Template (APST, 2002).

An agenda towards a Grid-service-oriented urban air-quality decision support system thus includes:

- Generating metadata for both atmospheric model components and data, to support future service discovery and selection: as Grid matures so applications will dynamically be configured – the metadata enables this. (Although it is arguably essential in its own right.)
- Enabling both real-time and archived air quality measurements as Grid/Web services
- An exploration of the atmospheric models and decision support environments to identify suitable decomposition into services
- Generation of parallelised services. (Grid middleware will not eliminate the need for parallelisation.)
-

The Grid technology is advancing quickly, as many prototype projects demonstrate, so that such activities should be undertaken with urgency if the potential of the Grid is to be fully explored.

CONCLUSIONS

The emerging technologies for Grid services are changing the technological landscape in which many environmental decision support systems will run. Rather than one program running on one machine, a new pattern will emerge, in which multiple services will be selected, configured and invoked on many different machines, exploiting multiple processors both within a service and by concurrent execution of services.

In this scenario software must be flexible, modular, configurable, and extendable at many scales:

- Within a service (e.g. a model) where additional or alternative modules can be inserted (for example to change the chemistry being modelled)
- Within a distributed "virtual computer" in which services can be made available by publication to registries, and discovered and run under the control of middleware.

In anticipation of this scenario we encourage model developers to utilise strongly structured methods, such that when their software is proven, the code can be repackaged as services suited to invocation from the Web or Grid.

REFERENCES

APST 2002 AppLeS Parameter Sweep Template http://grail.sdsc.edu/projects/apst/index.html

Cactus 2002, http://www.cactuscode.org/

Costanza, R., Gottlieb, S., 1998. Modelling ecological and economic systems with STELLA: Part II. Ecological Modelling 112, 81-84

Fedra K., 1999, Model-based Decision Support for Integrated Urban Air Quality Management, http://www.ess.co.at/docs/papers/fedra99.html

Foster, I., Kesselman, C. 1999, (eds.). The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann

Foster, I. Kesselman C., Tuecke S., 2001, The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International J. Supercomputer Applications*, 15(3).

Foster, I., Kesselman, C., Nick, M.N., Tuecke, S., 2002. The physiology of the Grid. An OpenGrid Services Architecture for Distributed Systems Integration. http://www.globus.org/research/papers/ogsa.pdf.

Global Grid Forum, 2002, http://www.gridforum.org/

HEAVEN, 2002, Healthier Environment through Abatement of Vehicle Emissions and Noise http://heaven.rec.org/

Maidment D., 2002 ArcHydro: GIS for water resources, ESRI Press

Mineter, M.J., Dowers, S., Gittings, B.M., 2000. Towards a HPC framework for integrated processing of geographical data: encapsulating the complexity of parallel algorithms. Transactions in GIS 4, 245-262.

Muetzelfeldt, R.I., Taylor, J., 1997. The suitability of AME for agroforestry modelling. Agroforestry Forum 8, 7-9.

SUTRA 2002, Sustainable Urban Transport, http://www.ess.co.at/SUTRA/

W3C, 2002 World Wide Web Consortium, http://www.w3.org/