

US DOE Program Announcement LAB 01-06
Proposal to the Office of Advanced Scientific Computing Research
DOE National Collaboratories Program

Collaboratory for Multi-scale Chemical Science

Sandia National Laboratories, Livermore, CA 94551-0969

Larry Rahn, Tel: (925) 294-2091, Email: rahn@sandia.gov - Institutional point of contact
Christine Yang, Tel: (925) 294-2016, Email: clyang@ca.sandia.gov
John C. Hewson, Tel: (925) 294-4973, Email: jchewso@sandia.gov
Carmen Pancarella, Tel: (617) 630-0316, Email: carmen@ca.sandia.gov
Wendy Koegler, Tel: (925) 294-4877, Email: wkoegle@ca.sandia.gov

Pacific Northwest National Laboratory, Richland, WA 99352

Jeff Nichols, Tel: (509) 372-4569, Email: jeff.nichols@pnl.gov - Institutional point of contact
Brett Didier, Tel: (509) 376-7965, Email: brett.didier@pnl.gov
Theresa Windus, Tel: (509) 372-6410, Email: theresa.windus@pnl.gov
James D. Myers, Tel: (610) 355-0994, Email: jim.myers@pnl.gov
Karen Schuchardt, Tel: (509) 375-6525, Email: karen.schuchardt@pnl.gov
Eric Stephan, Tel: (509) 375-6977, Email: eric.stephan@pnl.gov

Argonne National Laboratory, Argonne, IL 60439-4844

Al Wagner, Tel: (630) 252-3597, Email: wagner@tcg.anl.gov - Institutional point of contact
Branko Ruscic, Tel: (630) 252-4079, Email: ruscic@anl.gov
Michael Minkoff, Tel: (630) 252-7234, Email: minkoff@mcs.anl.gov
Lee Liming, Tel: (630) 252-5648, Email: liming@mcs.anl.gov
Sandra Bittner, Tel: (630) 252-0934, Email: bittner@mcs.anl.gov
Brian Moran, Tel: (630) 252-4021, Email: bmoran@anl.gov

Lawrence Livermore National Laboratory, Livermore, CA 94551

William J. Pitz, Tel: (925) 422-7730, Email: pitz1@llnl.gov

Los Alamos National Laboratory, Los Alamos, NM 87545

David R. Montoya, Tel: (505) 665-5675, Email: dmont@lanl.gov

NIST, Gaithersburg, MD 20899-8381

Thomas C. Allison, Tel: (301) 975-2216, Email: thomas.allison@nist.gov

MIT, Cambridge, MA 02139

William H. Green, Jr., Tel: (617) 253-4580, Email: whgreen@mit.edu

University of California, Berkeley, CA 94720-1740

Michael Frenklach, Tel: (510) 643-1676, Email: myf@me.berkeley.edu

Table of Contents

Abstract	v
1 Narrative	1
1.1 Background and Significance.....	1
1.1.1 <i>Multi-scale Dependencies – The Need for Collaboration</i>	1
1.1.2 <i>An Informatics Approach to Multi-scale Science</i>	1
1.1.3 <i>Combustion: A Multi-scale Chemical Sciences Pilot</i>	2
1.1.4 <i>Combustion Research across the Multi-scale Chemistry Community</i>	3
1.2 Preliminary Studies.....	3
1.2.1 <i>Molecular Science Software Suite (NWChem, Ecce, and ParSoft)</i>	3
1.2.2 <i>Active Thermochemical Tables</i>	4
1.2.3 <i>XML-based Data Standards at NIST</i>	4
1.2.4 <i>GRI-Mech</i>	5
1.2.5 <i>Feature Mining of DNS Data Sets</i>	5
1.2.6 <i>DOE2000 Collaboratory Development and Deployment</i>	5
1.2.6.1 Diesel Combustion Collaboratory.....	5
1.2.6.2 EMSL Collaboratory.....	6
1.2.6.3 Electronic Notebooks.....	6
1.3 Research Design and Methods.....	6
1.3.1 <i>Motivating Multi-scale Combustion Scenario</i>	7
1.3.2 <i>Architecture</i>	8
1.3.3 <i>MCS Community Portal</i>	8
1.3.4 <i>System Security</i>	10
1.3.5 <i>Knowledge Management</i>	10
1.3.5.1 Metadata/Annotation Management Overview.....	11
1.3.5.2 Community Standard Schemas.....	12
1.3.5.3 Resource Wrapping and Registration / Metadata-enabled Data Repositories.....	12
1.3.5.4 Multi-scale Integrated Search.....	13
1.3.5.5 Content Submission and Retrieval.....	13
1.3.5.6 Data Pedigree and Dependency Browsing.....	14
1.3.6 <i>Research Support</i>	14
1.3.6.1 Shared Document Repository.....	14
1.3.6.2 Electronic Notebook.....	15
1.3.6.3 Portal ↔ Domain Application Integration.....	15
1.3.6.4 Computational Gateway.....	16
1.3.7 <i>Community Interaction</i>	16
1.3.7.1 Notification.....	16
1.3.7.2 Chat/Discussion.....	17
1.3.7.3 Conferencing.....	17
1.3.7.4 Community Review.....	17
1.3.7.5 Access Management.....	18
1.3.8 <i>Applications and Data</i>	18
1.3.8.1 Community Databases.....	18
1.3.8.2 Information Sharing Across the Scales.....	19
1.3.9 <i>Outreach to the Scientific User Community</i>	21
1.3.10 <i>Management Structure</i>	21
1.3.11 <i>Timetable</i>	22
1.4 Subcontractor or Consortium Arrangements.....	24
1.4.1 <i>Project Team Consortium</i>	24
1.4.2 <i>Reaction Design</i>	24
1.4.3 <i>Collaborations with other proposed SciDAC projects</i>	25
2 Literature Cited/References	27

6	Description of Facilities and Resources	31
6.1	Sandia National Laboratories	31
6.2	Pacific Northwest National Laboratory	31
6.3	Argonne National Laboratory	31
6.4	NIST	31
7	Appendix	33
7.1	Appendix A: Combustion Modeling from the Molecular to Device Scale	33
7.2	Appendix B: Active Thermochemical Tables.....	37
7.3	Appendix C: Open Data Management Solutions for Problem Solving Environments: Application of Distributed Authoring and Versioning (DAV) to the Extensible Computational Chemistry Environment	41
7.4	Appendix D: Collaboration and Outreach to Other SciDAC Programs	55
7.5	Appendix E: EMSL Computational Facilities and Capabilities	57

Abstract

Rapid advances in computational hardware and software along with innovative experimental techniques are revolutionizing the rate at which chemical science research can produce the new information necessary to advance combustion technology, straining the traditional methods of communication through peer-reviewed literature and static databases. We propose to develop a pilot Collaboratory for the Multi-scale Chemical Sciences (CMCS) that will bring together leaders in scientific research and technological development across multiple DOE laboratories, other government laboratories and academic institutions to develop an informatics-based approach to synthesizing multi-scale information to create knowledge in the chemical sciences. The CMCS will use advanced collaboration and metadata-based data management technologies to develop an MCS (Multi-scale Chemical Sciences) portal providing community communications mechanisms and data search and annotation capabilities. The portal will also provide capabilities for defining and browsing cross-scale dependencies between data produced at one scale that is used as input for computations at the next. Notification mechanisms will make both researchers and their applications aware of updated values of relevant information such as reaction rates. The CMCS and its MCS portal will provide mechanisms to enhance the coordination of research efforts across related sub-disciplines in the chemical sciences, focusing research at one scale on obtaining or refining values critical in the next, reducing work performed using limited or outdated values, and enhancing the ability of the community to meet the national research challenges of DOE.

1 Narrative

1.1 Background and Significance

1.1.1 *Multi-scale Dependencies – The Need for Collaboration*

The area of chemical sciences is representative of many DOE programs in that it addresses complex multi-scale phenomena. The situation is similar in earth system studies, fusion research, high-energy physics, biology, and other areas of science – an understanding of environment and device scale phenomena requires more than simply applying one type of computation, with increased computing power, across scales. Different physical phenomena dominate system dynamics at these different scales, leading to a variety of models and experiments relevant in the different regimes. Information from one regime is used as input for the next, essentially “bootstrapping” from the atomistic to the device level. One of the major bottlenecks in such a multi-scale research enterprise is the passing of information from one level to the next in a consistent and validated manner.

The scientific process described above leads to a data- and model-centric view of the communications between sub-disciplines working at different scales. Data at one level is analyzed to develop a model that produces data used in turn by another, repeatedly across the range of scales and types of chemical information required. However, in this process more than just the raw data values need to be communicated. Confidence in a value’s accuracy, its uncertainty, dependencies on other data, etc. must all be considered when using it in further computational and experimental research. In the direction of decreasing length and time scales, information about the sensitivity of models on particular data may place a premium on very accurate values for certain fundamental quantities. Enabling the rich bi-directional exchange of both data and metadata between scales is a critical issue in making progress.

Traditionally, this information flow has been accomplished through the research literature and, more recently, through databases of chemical values. Discovery of new information in these sources is a manual process. Further, the information is fragmented. Determining whether results presented in a paper depend on obsolete values from a different regime may require searching through several papers and databases. These factors make communication difficult and time consuming and increase the likelihood of redundant and irrelevant research.

Across the domains represented in DOE’s Scientific Discovery through Advanced Computation (SciDAC), communication of expertise and the flow of information between sub-disciplines targeting different physical regimes will be as critical as increasing computational capabilities within a domain in effectively and efficiently producing practical results from basic research. Current manual approaches to coordinating multi-scale research cannot themselves scale to the amounts of data that will be generated through SciDAC and to the level of effectiveness and efficiency required to tackle national science issues in a cost effective manner. The multi-scale communications challenges facing SciDAC researchers are not discipline specific. Thus, a solution to these issues in the chemical sciences will provide a model for multi-scale science that can guide efforts in other domains.

1.1.2 *An Informatics Approach to Multi-scale Science*

To overcome current barriers to collaboration and knowledge transfer among researchers working at different scales, a number of enhancements must be made to the information technology infrastructure of the community:

- A collaboration infrastructure is required to enable real-time and asynchronous collaborative development of standards for data and metadata description, inter-scale scientific communication, geographically distributed disciplinary collaboration, and project management.
- Tools now used to generate and analyze data at each scale must be modified to enable generation and storage of the required metadata in a format that allows interoperability with other tools and collaborative functions, and must be made available for use by geographically distributed collaborators.
- Repositories are required to store chemical sciences data and metadata in a way that preserves data integrity and allows web access.
- New tools are required to search and query metadata, and to retrieve data across all scales, disciplines, and locations. These tools should be available via an integrated user-customizable interface or portal.

The complexities of managing information within such an infrastructure are daunting and the creation, communication and use of the additional information could quickly become unwieldy. However, recent technological advances, in particular the development of the extensible markup language (XML) [1] for defining machine and human

readable metadata based on standard schema, have significantly reduced the barriers to creating such a comprehensive informatics environment.

We propose a Collaboratory for Multi-scale Chemical Science (CMCS) focusing on combustion research that will demonstrate that an integrated multi-scale approach to scientific and engineering research is not only possible but can produce significant benefits in harnessing research to address real-world issues. The field of combustion is critical to the DOE mission for clean and efficient energy, and the DOE has ongoing investments in research across the full range of relevant scales and disciplines. The CMCS will bring an integrated, informatics-based approach to combustion research that enhances and begins to automate the flow of information between sub-disciplines.

CMCS efforts to develop tools supporting the multi-scale analysis of chemical systems for combustion will be directly applicable to other communities in the chemical sciences and related fields. Furthermore, CMCS will provide a model for multi-scale science that can be replicated in other domains.

1.1.3 Combustion: A Multi-scale Chemical Sciences Pilot

Fossil fuel energy supply and fossil-fueled combustion systems are the cornerstones of the industrial and commercial sectors of the U.S. economy, accounting for 85% of the energy consumed in the United States each year. In the private sector the combustion of fossil fuels provide a level of comfort and mobility for U.S. citizens that is unrivaled in the world. Fossil fuels continue to be inexpensive and the supply of fossil fuels remains stable, although heavy dependence on foreign sources has led to major economic and societal dislocations in the past twenty-five years and threatens to again in the future. Also, recent changes in international environmental mandates for lower CO₂ emissions have emerged as strong drivers for increased combustion efficiency. Despite continuing investments in alternative energy sources, the importance of hydrocarbon fuels, as they relate to the economy and quality of life in the United States, is unlikely to change in the foreseeable future.

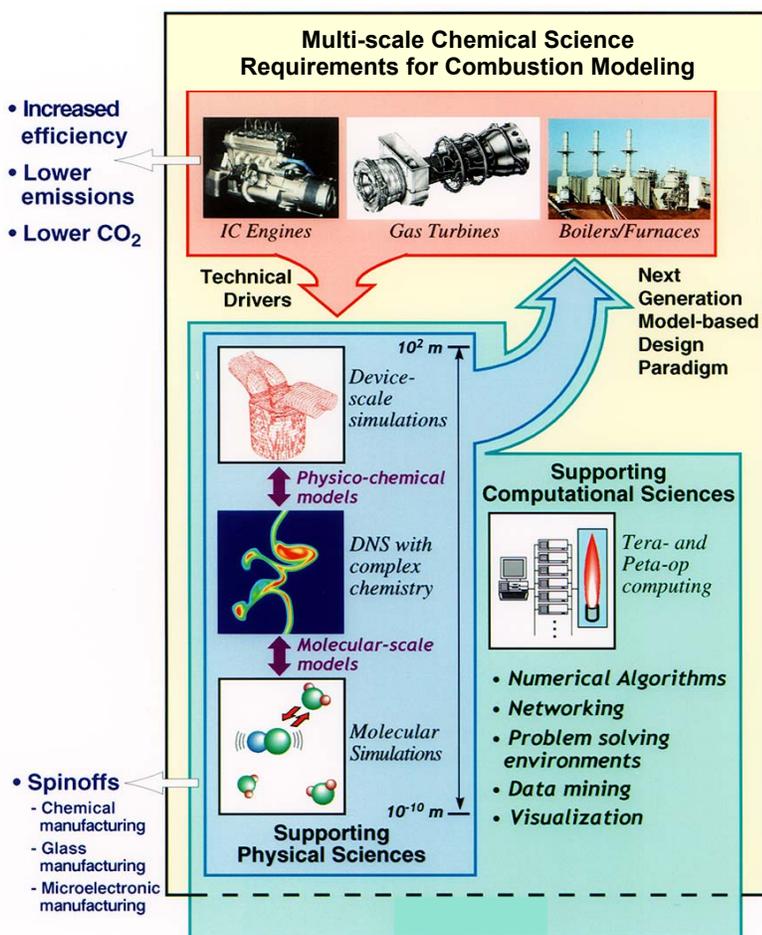


Figure 1. Combustion modeling requires the integration of scientific knowledge over a large range of scales

The advancement of the DOE mission for efficient, low-impact energy sources and utilization relies upon continued significant advances in fundamental chemical sciences and the effective use of the knowledge accompanying these advances across a broad range of disciplines and scales. This challenge is exemplified in the development of predictive computational models for realistic combustion devices. Combustion modeling requires the integration of computational physical and chemical models that span space and time scales from atomistic processes to those of the physical combustion device itself as illustrated in Figure 1.

Combustion systems involve three-dimensional, time-dependent, chemically reacting turbulent flows that may include multiphase effects with liquid droplets and solid particles in complex physical configurations. Against this fluid-dynamical back-drop, chemical reactions occur that determine the energy production in the system, as well as the emissions that are produced. For complex fuels, the chemistry involves hundreds to thousands of chemical species participating in thousands of reactions. These chemical reactions occur in an environment that is defined

by both thermal conduction and radiation. Reaction rates as a function of temperature and pressure are determined experimentally and by a number of methods using data from quantum mechanical computations. The collaborative creation, discovery, and exchange of information across all of these scales and disciplines are required [2] to meet DOE's mission requirements. [Chemical sciences relevance to combustion is discussed further in Appendix A.]

1.1.4 Combustion Research across the Multi-scale Chemistry Community

Combustion research is a relatively mature discipline, having adopted a coordinated approach to the development of chemical models to a greater degree than some other areas within the chemical sciences. A variety of databases and community models have been developed. The CMCS brings together leaders in the chemical sciences that have developed key community resources:

- **Molecular science simulations** characterizing potential energy surfaces, electronic structure, and the dynamics of atoms, molecules and clusters using quantum chemical methods. (PNNL)
- **Active tables** for the analysis of thermochemical properties of systems of molecules and description of the metadata documenting their computation from more fundamental measurements and computations (ANL).
- **Kinetic reaction rate parameters**, including the transition-state theory tools for computing and estimating uncertainty, and the metadata involved in their development, evaluation and cataloging (NIST, LLNL, UCB, MIT and SNL).
- **Chemical kinetic mechanisms, including** tools to develop, validate and reduce them (NIST, UCB, LLNL, MIT, and SNL).
- **Feature mining and analysis** of direct numerical simulation (DNS) data for flame structures and characteristics and the tools to visualize and analyze DNS data features to validate submodels of combustion processes. (SNL).

The resources described above are located at major centers of research within the DOE Chemical Sciences program and are associated with critical-mass efforts that offer long term benefits. Some are associated with Collaborative Research Center roles with facility funding, enabling them to promote capabilities established by a pilot project within the larger community and contribute to their maintenance and continuing development. While other community resources exist, these elements are representative and span the regimes relevant to combustion research. Addressing the flow of information from validated physico-chemical submodels into next-generation model-based design simulations is outside the scope of the currently proposed effort, but could be addressed through follow-on activities. Coupling this effort to others such as the Diesel Combustion Collaboratory [3], would then complete the 'bootstrapping' process to the benefit of actual combustion devices.

1.2 Preliminary Studies

In order to develop the CMCS, we draw on a variety of personnel with experience in both scientific research and the development of collaboratory technologies. Members of the chemical sciences research community are drawn into the project on the basis of their motivation to develop a new research paradigm based on greater collaborative efforts. Their experiences provide the architectural direction for the CMCS. Other team members are drawn from the DOE2000 Collaboratory program and the developers of the Molecular Science Software Suite (NWChem, Ecce, and ParSoft) for their experience in developing and implementing the relevant middleware technology and infrastructure. The preliminary studies summarized here are indicative of both expertise in the full range of scales and disciplines supporting combustion science, and expertise in the required areas of computer and information science.

1.2.1 Molecular Science Software Suite (NWChem, Ecce, and ParSoft)

NWChem [4-6] is a new generation of high-performance molecular modeling software that runs on parallel computing systems ranging from clusters of workstations to the emerging teraflop class of massively parallel computers. It provides a broad range of capabilities for solving sophisticated mathematical models of chemical systems from first principles at both the molecular orbital and density functional theory levels. These capabilities enable theoretical chemists to predict the fundamental characteristics of chemical systems at a level of accuracy that is otherwise obtainable only from the most sophisticated experimental approaches. NWChem also supports molecular dynamics calculations with a variety of empirical as well as quantum mechanical force fields to simulate macromolecular and solution systems. The software is modular, making extensive use of object-oriented design concepts. The object-oriented design of NWChem will facilitate its use within the CMCS since well-defined integration points already exist.

Ecce (Extensible Computational Chemistry Environment) [7] is a domain-encompassing problem-solving environment for computational chemistry composed of a suite of distributed client/server UNIX-based X Window System applications seamlessly integrated. The resulting environment includes tools to assist the user with many tasks, including management of projects and calculations, construction of complex molecules and basis sets, generation of input decks, distributed execution of computational models, real-time monitoring, and post-run analysis. Ecce was developed by PNNL and has been operational since 1997. Ecce as part of the Environmental Molecular Sciences Laboratory Molecular Science Software Suite won an R&D Magazine award in 1999 for being one of the 100 most significant technological innovations of the year and a Federal Laboratory Consortium award in 2000 for excellence in technology transfer. More details about this software suite can be found in Appendix E.

Ecce is currently completing modifications to redesign its data model, adopt a protocol-based interface to the underlying data storage (removing its dependency on the underlying object-oriented data base management structure), and providing layered software interfaces as abstractions for integrating applications. Ecce now provides an unprecedented level of access to its data store, leading to a variety of possibilities that were previously not available to collaborators. The combination of the Web's Hyper Text Transfer Protocol (HTTP) [8], XML, and the Distributed Authoring and Versioning (DAV) protocol [9] were chosen as implementation technologies due to their closest conceptual mapping to Ecce's design goals. The overall goals of this effort and the implementation details are described in a draft paper in Appendix E.

As part of ongoing research into generalized components for problem solving environments, PNNL has designed and prototyped a web-based architecture for submission and monitoring of computational jobs. The design was implemented as a portal service that can be accessed by applications written in Java, C++, or any browser supported language. The portal service accepts XML job requests over an HTTP connection, securely moves files to the target compute server, submits the job to the queuing service or operating system, and provides information about the job state to users or client applications. PNNL also developed a passive monitoring capability in which the job request may include an XML description of the application output data and be used to provide a dynamic, real-time feed of user-selected data back to the user. This effort has provided experience with and insights into a number of technologies and issues relevant to the CMCS proposal including: security requirements of web-based portal architectures, loosely coupled component design based on protocols and XML, requirements of data grid services, and the role of resource information and resource discovery services within such architectures.

1.2.2 Active Thermochemical Tables

Measurements of enthalpies of formation of a chemical species are always relative to other species, via enthalpies of reactions, bond dissociation energies, etc. Similarly, calculations of properties from first principles (*ab initio*) methods, are generally most accurate when considering properties relative to other similar species; in these cases properties are often calculated by taking advantage of isodesmic (involving the same number and type of chemical bonds) reactions. The result is that properties are best described by their relationship to other properties in a database. Furthermore, uncertainties in the derived elements of the property databases often combine with each other. Active tables (which are described in more detail in Appendix B) deal with this issue by providing a computational representation of the relationships between molecular-scale data and derived thermochemical properties of molecules. As such, active tables will allow the CMCS to capture these relationships as metadata and thus enhance the communication of molecular-scale data to higher scales.

Dr. Ruscic and collaborators at ANL have recently demonstrated [10, 11] the usefulness of the active tables concept on two small pilot thermochemical networks. This approach alleviates the hidden inconsistencies and difficulties in traditional methods for introducing and propagating new primary data in a global way to determine thermochemical properties of molecules. As opposed to the traditional approach, the use of a thermochemical network allows the achievement of an optimal global solution to the desired thermochemical properties for all species involved. The global solution is simultaneously consistent with all available underlying relationships. In addition, through a statistical analysis of the network, the thermochemical network approach allows the identification of initial relationships that were either incorrectly determined or have too optimistic confidence levels. This statistical analysis results in proposing approaches to alleviate such inconsistencies, converging to a self-consistent network. Finally, the network approach allows for a fast and easy update of the global simultaneous solutions when new basic data becomes available.

1.2.3 XML-based Data Standards at NIST

Two proposals for the development of XML-based data standards were recently funded at NIST. One of these projects is led by a CMCS co-investigator (Allison). The initial stage of the development corresponds closely to

several of the collaboratory goals for data formats for thermodynamic and kinetic data. Later stages of the proposal focus on the development of tools, repositories, and data sharing. It is intended that this development be compatible with the collaboratory to the greatest extent possible. Long-range goals for the NIST effort call for ratification of the XML data formats by a standards organization and for working with journals to accept data in the XML format. There is an emphasis on free and widespread distribution of the XML format and software tools at all stages of the effort. NIST will benefit greatly from the contributions from members of the CMCS project and other members of the chemical community to the effort of establishing standard data formats. Participation by a large and diverse group will enhance the likelihood of broad acceptance by the community.

1.2.4 GRI-Mech

A new paradigm for developing models of complex chemical systems has been demonstrated for natural gas combustion under the auspices of the Gas Research Institute. It is an excellent example of the research that CMCS seeks to enable. That model, known as GRI-Mech, has been developed by a team of researchers from four different institutions using the method of "Solution Mapping" developed by the University of California, Berkeley participants [12-13]. In brief, after examining available databases for all relevant rate and thermochemical parameters and assuring, as a collaborative group, that they were in accord with modern theory, collaborators sought information on experiments in the literature, or performed experiments, that were sensitive to many of these parameters. Each of the experiments was then modeled with a trial mechanism using Chemkin-based tools [14]. The combined results of this modeling were then fit by a set of polynomials using a factorial design method. The differences between experiments and polynomial fits are minimized by systematic variation of the sensitive variables, which were constrained to lie within a prescribed experimental error range, with the constraints based on collaborative discussions among the various team members. The entire process was carried out across the web using daily email and ftp exchanges. The outcome is a model for natural gas combustion that accurately represents the consensus view of a larger collaborative team than had previously developed mechanisms. This mechanism can be used as an aid for design of natural gas combustors. This effort also explored the use of the Chemical Markup Language (CML) [15] and has written parsers to convert Chemkin formatted mechanism to XML. GRI-Mech is described on the web at: http://www.me.berkeley.edu/gri_mech

1.2.5 Feature Mining of DNS Data Sets

Recent improvements in the Sandia DNS code (S3D) for compressible reacting flow [16] with detailed chemistry, and the availability of 'shake-down' time on ASCI 'Frost' have enabled the development of 0.75 TByte of hydrogen-air autoignition data. A newly developed tool for identifying and extracting features in time-varying multidimensional datasets, FDTOOLS, has also been developed. FDTOOLS is written in the Common Component Architecture [17] and is a prototype for data analysis and mining tools for SciDAC DNS data sets produced by the Computational Facility for Reacting Flow Science. These tools identify and extract of features corresponding to turbulent reacting flow phenomena which can then be used in further analysis, model validation, and model development.

1.2.6 DOE2000 Collaboratory Development and Deployment

The DOE2000 Collaboratory program combines research in Internet-based collaboration technologies and distributed group interaction with pilot collaboratory deployment in a variety of scientific communities. CMCS partners have been involved in all aspects of Collaboratory development, producing innovative technologies and successfully deploying them to support the needs of distributed research teams.

1.2.6.1 Diesel Combustion Collaboratory

The Diesel Combustion Collaboratory (DCC) [3] was a DOE 2000 pilot project to develop and deploy collaborative technologies to combustion researchers distributed throughout the DOE national laboratories, academia, and industry. The DCC project team was geographically distributed across DOE laboratories and universities. To build the DCC, the project team set up electronic notebooks, shared document archives, and employed both asynchronous and synchronous collaborative tools for project management and project meetings. The resulting collaborative problem-solving environment for combustion research enabled the remote execution of combustion models on geographically distributed computers, the ability to archive and share experimental or model data, the use of electronic notebooks and shared workspaces for facilitating collaboration, and the seamless integration of modeling tools, visualization tools, and data storage tools in a secure manner across the Internet. From the DCC portal (<http://www-collab.ca.sandia.gov/>), users have secure access to web resources, data archives and modeling codes

using certificates and Akenti security [18]. The data archives, resources, and modeling codes are located at widely distributed sites. These codes include Chemkin and HCT (a related LLNL capability) codes, both used by researchers in the validation of chemical mechanisms. From a single portal, all DCC collaboration tools and data resources are available to users with security credentials. Experience in the DCC agrees with industry case studies showing the importance of such seamless integration, and access control in increasing the effectiveness of distributed teams.

1.2.6.2 EMSL Collaboratory

PNNL researchers have developed a real-time collaboration environment [19], and an electronic laboratory notebook [20], and have investigated a variety of collaboratory architectural issues under DOE2000. A participatory design process has led to successful production deployment of these and other technologies within PNNL's Environmental Molecular Sciences Laboratory [21].

A variety of internal laboratory projects have also investigated concepts and technologies relevant to CMCS. These include the development of a lightweight web-portal-based job launching mechanism, XML-based resource discovery capabilities, XML-based metadata management and generation tools, and an investigation of generic binary-XML-binary file translation capabilities using the eXtensible Scientific Interchange Language (XSIL) [22].

1.2.6.3 Electronic Notebooks

We anticipate substantial use of Electronic Notebooks within this collaboratory for management, documentation, and research purposes. PNNL, through its involvement in the DOE2000 Collaboratories program has been a lead institution in the development and deployment of Electronic Laboratory Notebooks (ELN). Over 600 researchers and developers from around the globe have downloaded the PNNL ELN since early versions of the software were first made publicly available in 1995. A variety of research projects at sites including PNNL, the National Center for Atmospheric Research (NCAR), and Columbia University, routinely use the ELN as part of their research process, and actively communicate new requirements and suggestions to the ELN team. PNNL's experience with notebook development and deployment, and anticipated involvement in the development of next-generation notebook technologies, will be a valuable asset in understanding how to best utilize notebooks to satisfy CMCS objectives.

1.3 Research Design and Methods

To demonstrate the effectiveness of a community-based informatics approach to multi-scale science, we propose a Collaboratory for Multi-scale Chemical Science (CMCS). CMCS will provide mechanisms to enhance and automate the flow of scientific information and collaboration across scales through community models and databases in support of combustion research. The design of CMCS capabilities will be guided by an overarching scenario connecting the results of quantum mechanical calculations and experimental measurements of atomistic properties to their effects observed in the features mined from the results of DNS of turbulent reacting flows. This scenario is detailed in Section 1.3.1. Chemistry researchers and information technologists will work together to target CMCS efforts towards eliminating the communications bottlenecks outlined in the scenario and towards demonstrating the value of CMCS in combustion research. While CMCS capabilities will be available to the broad combustion community and be designed such that additional resources may be added, the scope of this pilot project will be limited to the specific set of community resources (models and data) identified in the scenario. By design, the resources required are already a priority focus within the chemical science programs at the participating institutions, freeing CMCS to concentrate on inter-scale collaboration and integration issues.

The ultimate goal of combustion research is to apply chemical knowledge to the construction of clean and efficient energy sources. In the CMCS pilot project, this end goal is represented by the validation of models for turbulent reacting flow. Modeling the performance of engineering devices relies directly on such subgrid scale models to describe the effect of turbulent mixing on chemical reactions in the device. In the following section, details are given on the flow of information in our scenario. CMCS is motivated by the requirements to support the round-trip collaborative information flows within this process from the quantum scale up to reacting flows and back again to smaller scales. More detailed information on the scope of combustion research can be found in the appendix and references. Following the scenario, sections 1.3.2 – 1.3.7 detail the technologies that will be developed and deployed within CMCS. Section 1.3.8 returns to the scenario and describes how an initial group of combustion researchers will use the resources and capabilities described to conduct a coordinated multi-scale combustion research effort as a demonstration of CMCS's potential to the larger community.

1.3.1 *Motivating Multi-scale Combustion Scenario*

The scenario leads to anticipated impact on the DOE mission to provide efficient, low-impact energy sources. This mission motivates research efforts across universities and DOE laboratories spanning fundamental molecular science to industry collaborations targeting development and validation of computational design tools.

Fundamental properties of ignition-relevant molecules are computed by a scientist remotely using massively parallel molecular simulations. Ionization energies, molecular electronic potential energy surfaces, vibrational frequencies, and molecular dissociation energies are obtained and annotated with information regarding their remaining uncertainty, method of computation, and relationship to other data. A paper is drafted and submitted for publication. These data, along with similarly documented experimental data are stored in an archive(s). The availability to another researcher of these quantities allows her to discover that this is almost enough data to enable the computation of derived thermodynamic properties for an important chemical species. Through collaborations with colleagues, the required remaining properties are computed, and the new thermochemical data is derived.

These new data are documented for their dependencies on other data, uncertainties derived from these dependencies, and consistency with other computational and experimental results. Uncertainties also propagate between dependent values. An understanding of these relationships enables identification of sets of thermochemical values that must be updated, through another collaborative process, to produce consistent improved thermochemical data. Access to the data and documentation enables workers at NIST to discover the data, evaluate it, and determine that the pedigree of this data meets their standard for publication in a public archive. The derived thermochemical data is thus also discovered, and similarly added to the public archive.

The availability of data for many ignition-related species and reactions from multiple archives enables another scientist to propose a chemical mechanism for the autoignition of a new fuel. The mechanism involves hundreds of chemical species participating in thousands of reactions; many of the reaction rates are estimated using simple techniques. In a multi-person collaboration, the mechanism is validated against many different sets of experimental reaction data. By adjusting the input reaction rates within defined limits set by the data pedigree, the collaborative team defines and documents the range of validity of the reaction mechanism for the new fuel. The mechanism does not predict certain ignition experiments within the desired degree of accuracy, and the scientists conduct a sensitivity analysis for those conditions. Certain reaction-rate parameters are found to have high sensitivities and high uncertainties, leading to significant uncertainties in the ability to predict ignition experiments. The scientists initiate a collaborative project with a group of chemists to refine those parameters that had previously been estimated. A series of experimental and computational studies ensue. The same molecular scientist who submitted a paper above is involved in this collaborative process to study a transition state before the submitted paper even shows up in print. The refined parameters improve the prediction of the ignition experiment, and the scientists make the mechanism available to other researchers. NIST researchers receive automatic notification of the mechanism and update the pedigree of the reaction rate data in the public archive.

A combustion researcher interested in developing an autoignition submodel becomes aware of the mechanism, but finds that the cost of the desired direct numerical simulation (DNS) is prohibitive with the hundreds of species present in the mechanism. This researcher initiates a collaborative effort with another researcher who creates a reduced mechanism with the desired number of species, documents its range of validity, and provides the reduced mechanism in a compatible format. Together these researchers use the DNS to study the chemical response of the system to turbulent mixing. They identify the conditions leading to the earliest ignition with feature tracking tools. These features are documented and archived, and this information is used to improve ignition models. Another series of DNS runs indicate inhibited ignition; the relevant features are again documented and archived. A comparison between the DNS initial conditions in each case furthers the knowledge of ignition limits. This information is linked to the combustion model for further reference by device-scale modelers.

After several iterations of the above process, the mechanisms and submodels develop well-defined pedigrees and are suitable for inclusion in advanced device-design simulations. One significant issue here is the time for information generated at different scales to make their way to other researchers who need that information. If the entire process waited for publications at each stage, of which there are perhaps five intermediate ones, the time for the results of a molecular simulation to make impact on a reduced mechanism and a combustion model would be close to one decade. The CMCS will pilot the necessary infrastructure and methods to decrease this time-to-impact.

1.3.2 Architecture

The description above provides a brief glimpse of the inherent complexity that must currently be managed manually by combustion researchers. CMCS does not seek to eliminate this complexity, but to provide community-wide capabilities for managing it. As shown in Figure 2, CMCS will provide the majority of its capabilities through a web-based Multi-scale Chemical Science (MCS) portal. This portal will provide a customizable access point for accessing community knowledge bases, interacting with other members of the multi-scale chemistry community, performing and documenting experiments, and supporting the multiple paths of information flow necessary to integrate these activities. These capabilities rely on an underlying Grid [23] infrastructure including a sophisticated metadata and annotation management subsystem. Standard chemical information interchange schema combined with metadata/data translation mechanisms enable loose federation of underlying group and community data stores and global tracking of information such as data pedigrees. Chemistry domain applications, modified or extended to understand the community-developed schema, help automate the collection of and directly exploit data pedigree and other newly available data annotations.

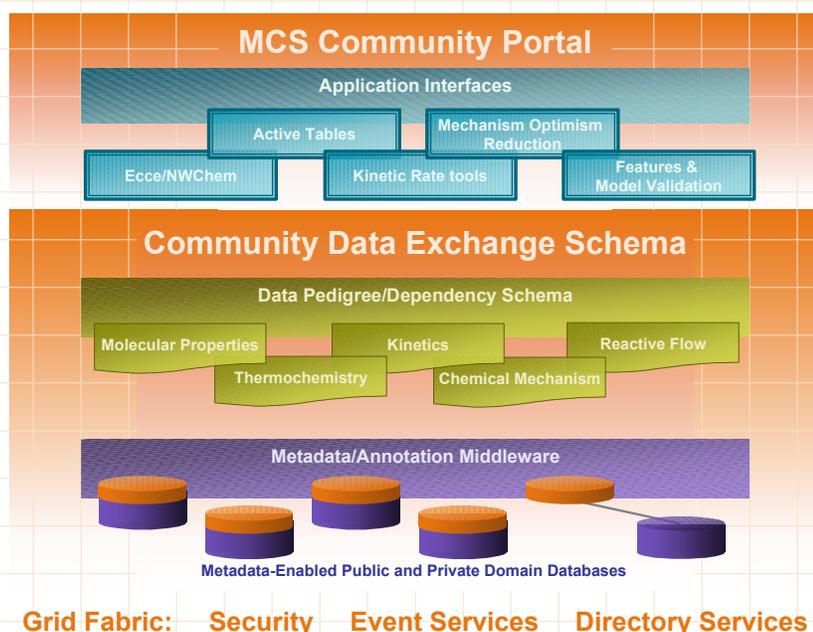


Figure 2. Architecture diagram for CMCS showing portal integration of domain applications and data resources from across the multi-scale chemistry community.

1.3.3 MCS Community Portal

The multi-scale chemical science (MCS) portal will be developed as the focal point of the collaboratory. As shown in Figure 3, the portal will provide a broad range of capabilities. Bringing these capabilities together across chemistry sub-domains will provide convenience while helping researchers think and act in the larger combustion context. For the purposes of discussion, portal capabilities are divided into groups related to accessing community knowledge resources, interacting within the community, and generating new knowledge through research projects. These activity-based groupings are clearly not orthogonal and our design does not make any such distinctions. Rather, the design prescribes the use of common services across all portal tools to enable facile communications between them, with the goal of enhancing researcher's ability to organize and move information through the various activities beyond what is currently possible.

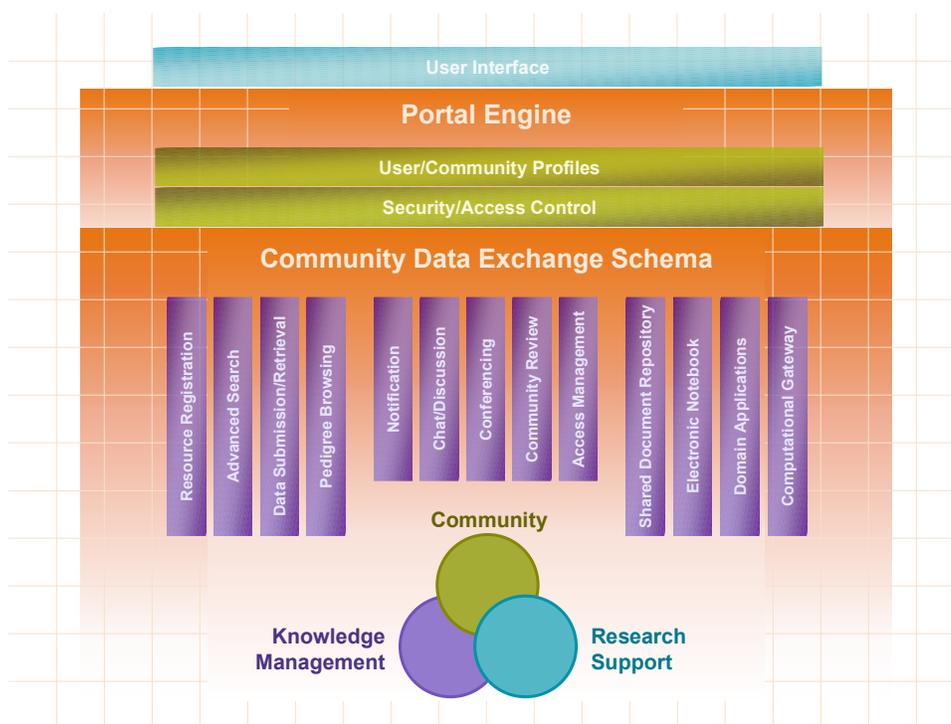


Figure 3. Prototype Design of Portal for Multi-scale Chemical Science.

Figure 3 shows details of the portal’s overall design. Users will be able to create profiles that define their preferred view(s) of the portal. Profiles can be used to aggregate users into communities of interest such as “Turbulent reacting flow model development” or “XML schema for reduced chemical mechanisms”. Security is integrated into the portal architecture, providing authentication, authorization control, and encryption capabilities that can be passed to underlying resources.

The MCS portal will be customizable so that it supports the needs of the user and the user’s workflow. It will allow users to emphasize information relevant to a subset of chemical scales or to emphasize data submission and search capabilities, tools for organizing and understanding relationships in the data, or the capabilities for interaction with colleagues while helping maintain contextual awareness of the overall community. Users will have access to portal configuration tools to create custom views and save them in personal, group, or community profiles.

A variety of web-based technologies exist for the development of such a portal. The WorkTools system [24], developed at the University of Michigan, is currently in use within the Space Physics and Aeronomy Research Collaboratory (SPARC) as a means of accessing data and colleagues. It allows researchers to select a set of tools such as chat boxes and live instrument data feeds and position them as desired within a persistent web page. Similarly, NCSA’s OPIE system [25] allows surfers to arrange tools within a browser window through the same click-and-drag operations used to arrange windows on a computer desktop. Within the DOE community, work has been done to allow Grid resources to be accessed through portals as exemplified by the NPACI HotPage site [26], and additional work to develop standards for portal layout via a PortalML language [27] are anticipated within SciDAC. Commercial tools for portal development also exist, targeted at companies wishing to create “My Yahoo” style enterprise portals for employees and customers. While each of these systems has strengths and weaknesses, many could provide the basic capabilities needed within MCS. In developing the portal, we will evaluate existing and emerging portal development environments, select one, and adapt it as needed to meet MCS requirements, potentially in collaboration with the environment’s developers.

Once the basic infrastructure is designed, we will begin to integrate specific MCS capabilities. Since significant technology development will be required to achieve the desired levels of functionality and integration, we will follow an incremental approach to developing the overall portal, using it to enable access to prototype capabilities very early in the project. Over time, more integrated and feature-rich versions of the individual tools will be developed and made available through the portal. To prioritize efforts and ensure overall system usability, specific

use cases based on the CMCS guiding scenario will be developed early in the project and periodically updated to reflect the growing understanding of research.

1.3.4 System Security

Due to its central role in CMCS, the portal is a logical place to coordinate system security. We envision the portal providing single sign-on capabilities across the tools represented. The wide range of tools to be integrated and the bi-directional flow of information between private and public repositories proposed in CMCS (as detailed in subsequent sections) make system security a challenging issue. Obtaining sufficient expertise with security technologies and in the design and deployment of secure services in distributed environments was a primary consideration in the selection of CMCS development team members.

Existing public key infrastructure (PKI) technologies can be leveraged to provide the basic authentication, authorization, encryption, and non-repudiation services required in CMCS. Because the portal will be web-based, standard web mechanisms for authenticating users, including using public-key certificates, can be applied. Web browsers and servers currently incorporate the technology necessary to request and validate user credentials and to set up secure socket layer (SSL) encrypted communications. The Grid Security Infrastructure (GSI), implemented, in collaboration with others, by the Distributed Systems Laboratory (DSL) at Argonne National Laboratory [28] is a PKI-based system that includes delegation capabilities, allowing a globally trusted public-key identity to be used to securely obtain a local credential in a non-PKI system, e.g. a UNIX username/password. A SciDAC proposal, "Security and Policy for Group Collaboration" [29] led by ANL, offers advancements in GSI that will be useful in latter stages of CMCS development. In CMCS, GSI would allow an MCS portal user's PKI credentials to be used to automatically obtain credentials for back-end systems such as databases. The advanced attribute-based Akenti system, which is used within the DOE2000 Diesel Collaboratory project, is intended to provide scalable security services in highly distributed, collaborative, multi-institutional environments. The work described in LBNL's "Distributed Security Architectures: Middleware for Distributed Computing" submission to SciDAC [30], which proposes to integrate GSI credentials and Akenti and also to integrate Akenti access control with the Distributed Authoring and Versioning (DAV) extension to the HyperText Transfer Protocol (HTTP) would provide significant benefits to CMCS. (The significance of DAV in the CMCS is discussed later in the proposal.) Finally, digital signature and timestamping services are also becoming readily available through software development kits and Internet services.

Thus, it should be possible to assemble a strong basic security infrastructure for the MCS portal from existing components through careful integration and deployment efforts. However, two aspects of CMCS may require advances to meet community needs. The first deals with limiting the amount of resources that can be used on the user's behalf when the portal delegates the user's credentials to a back-end or external system. Similarly, the length of time over which such delegation is allowed to occur may also need to be limited. Appropriate policies for various CMCS use scenarios will need to be defined and technologies to implement them must be developed. A variety of approaches can be used to provide incremental capabilities in this area, and we anticipate the development of general limited delegation capabilities within the Grid community. We will seek to collaborate with any such efforts by providing requirements and testing the systems developed. The second aspect of CMCS security requirements that will need to be refined during the project relates to the capabilities provided for attaching private annotations to publicly available data. The general solution of providing fine-grained access controls on each such relationship is clearly too cumbersome. Defining an appropriate level of access granularity to balance ease-of-use with user requirements will require experimentation during the project lifecycle. This issue is discussed in additional detail in section 1.3.7.5 after more of the CMCS infrastructure has been described.

1.3.5 Knowledge Management

CMCS will provide a unifying interface to combustion-related chemical science information. This information is currently scattered in public and private databases, flat files, and in the scientific literature, and is organized primarily along sub-disciplinary lines. Information is locked in a variety of formats and provided with varying degrees of validation and context. The NIST databases described in section 1.2.3 and [31] have been widely praised by the combustion community for their organizing influence, but the fraction of data contained therein is small. In CMCS, we propose to leverage the NIST and other databases, and provide added value to working researchers in a number of directions:

- Providing a single location from which to access the wide range of data needed for combustion research
- Reducing or eliminating the difference in access protocols and procedures between resources
- Developing community standards for representing information that crosses domain boundaries

- Using these definitions, combined with translation capabilities, to allow federated searches across multiple databases
- Using similar mechanisms to automate the reformatting of information retrieved from databases for use in research applications and vice versa
- Developing machine-processable representations of data pedigree and dependency relationships
- Providing tools for manual and automated traversal of pedigree and dependency relationships to support group and community data validation efforts and exploration of sensitivity analyses
- Allowing arbitrary annotations and comments to be attached to data values to support community expertise building and consensus formation activities

The underlying architecture proposed to address these issues, and the implementation plans for providing search, retrieval, and submission capabilities and for managing and exploring data pedigree information are detailed in the following subsections. Additional capabilities related to generation and exploitation of data annotations, and to the refinement and validation of information through community processes are discussed in sections 1.3.6 and 1.3.7 respectively. An initial exploration of the type of architecture proposed here, and a further discussion of its benefits, is reported in a draft paper in Appendix D.

1.3.5.1 Metadata/Annotation Management Overview

Metadata is commonly defined as information ‘about’ data values and data sets. However, such a definition is very dependent on one’s perspective. For example, whether a chemical formula is metadata about a molecular geometry, or whether geometry and other information such as the heat of formation are metadata about a chemical formula is a matter of perspective. Such differences of opinion, once encoded in software, are an endless source of barriers to cross-scale collaboration. Within this proposal, we modify this definition to equate the term metadata with data values that have meaning across domains. In the example above, the chemical formula is metadata because it has meaning for both quantum- and thermo- chemists. Heat of formation would not be considered metadata until we expand scope and realize that both thermochemists and kineticists ascribe meaning to it.

Although this shows that our definition continues to have some context-dependence, it is more powerful than the original. By our definition, data is opaque and meaning-free outside a sub-discipline and, as a corollary, efforts to standardize formats and meanings between collaborators to support inter-scale search capabilities, application interoperability, etc. can be confined to metadata. Further, the system architecture can treat data as opaque as well and no restrictions need be placed on its format. In contrast, because metadata must be understood and manipulated, it must be formatted in a way that exposes its meaning in machine-comprehensible form. An important consequence of this bifurcation is that it minimizes the effort required to allow two parties to collaborate – no changes are required to any applications, and no agreements need be reached about the meaning of terms, except those directly concerning the values that will be exchanged.

Such an architecture is at the heart of the DOE2000 electronic notebooks and is the reason they can be easily extended to handle new annotation types; the complexities of handling the annotation data can be confined to the components that create and render the annotation. No translation of the annotation data is required by the notebook and the base notebook system only assigns meaning, and has code to manipulate, metadata such as author name, creation date, data type, digital signatures, etc. In defining the CMCS architecture, we looked to exploit the architecture explored in DOE2000 electronic notebooks while taking advantage of technologies that have matured since their inception. The directions being proposed for the evolution of electronic notebooks and the broader concept of metadata-based system design proposed within the “Scientific Annotation Middleware (SAM)” [32] submission to SciDAC have been a key influence.

The basis for both CMCS and SAM architectures is the formatting of metadata using the XML [1]. XML is a powerful language for encoding the definition of technical terms in a human- and machine-readable form. XML’s expressive power, together with the availability of technologies for manipulating it – authoring, parsing, validating, translating, etc., have made it a de facto standard for information exchange in new systems.

Since both efforts will leverage the Web-based DAV protocol [9], CMCS anticipates being able take advantage of work done under the SAM proposal. DAV is an Internet Engineering Task Force (IETF) standard set of extensions to the HTTP/1.1 protocol to support basic data management over the web including storage and retrieval of typed, opaque data files/objects, content locking, hierarchical collections and annotation of the data with arbitrary metadata [9]. It defines the formatting of metadata in properties consisting of XML key:value pairs and provides operations for creating, removing, and querying them. The extensible DAV Searching and Locating (DASL) protocol [33] adds

methods for server-side search capabilities. It provides a basic search grammar and can be extended with additional grammars, e.g. XML Query.

A layered set of services built on top of DAV/DASL that provide successively more specialized capabilities has been outlined in the SAM proposal. The brief description here highlights functionality directly relevant for CMCS. The Metadata Management Services (MMS) will support simple federation of DAV servers, allowing propagation of storage and retrieval requests and queries down through a hierarchy of servers. It will also provide mechanisms for registering metadata generation tools that can parse data of specified types and generate new DAV properties. The SAM Semantic Services (SS) adds a standard for representing semantic relationships between DAV objects. A Notebook Services (NS) layer defines records management capabilities in terms of specific semantic relationships and property definitions. A set of interface components, including a graphical relationship browser, and programming interfaces provides access to the services. A SAM-based notebook will be built from these components that, as discussed in section 1.3.6.2, could be leveraged by the CMCS to develop novel functionality.

1.3.5.2 Community Standard Schemas

This high-level description of the proposed CMCS metadata management infrastructure provides the context necessary to discuss the tasks that will be undertaken in CMCS to provide community knowledge management capabilities. The critical initial steps are the standardization of definitions for relevant chemical information that will be exchanged by CMCS researchers and the formal representation of these definitions in XML. Such representations are referred to as schema. CMCS researchers will engage their respective chemical science communities in these schema development efforts through the developing MCS portal and traditional community forums. Although we do not wish to under-represent the effort that will be required, it should be noted that overall scope is relatively small. CMCS does not require a single, global schema that represents all of chemical knowledge. Only information that will be exposed as metadata need be defined. Further, schema can be evolved within the system and variants can be accommodated. Thus, for example, if the stakeholders for two community resources wish to define metadata quantities beyond the minimal set on which agreement can be reached, annotations can be supported in both dialects. If complete agreement can be reached at a later date, a translator can be registered with the system to allow queries expressed in the new schema to match metadata defined with the old standards.

As shown in Figure 2, we anticipate the definition of five potentially overlapping schema within CMCS that define the information that must be exchanged across the boundaries between the five chemistry sub-domains in the guiding scenario as well as a general schema for representing dependency data. The inter-domain schema will most likely leverage nascent community efforts such as the Chemical Markup Language (CML) [15], which defines a set of basic molecular properties such as chemical formula and molecular geometry. As noted in sections 1.2.1 and 1.2.3, efforts at PNNL and NIST to define quantum chemistry, and thermochemistry and kinetics information, respectively, can also be leveraged. If significant overlap is observed in the developing schema, it may be possible to pull common elements into a broader chemistry schema. Since there are no limitations in DAV and XML in using multiple schemas to define properties on a single object, any such adjustments can be made with little impact on CMCS scope.

Since the definition of dependency relationships is not chemistry-specific, it may be possible to form a broad collaboration amongst problem solving environment developers to define the concepts necessary to track data pedigree and map the sensitivity relationships that relate the uncertainty in one quantity to the uncertainty in derived quantities. Such a common schema would enable re-use of tools for traversing such relationships in multiple projects.

1.3.5.3 Resource Wrapping and Registration / Metadata-enabled Data Repositories

DAV is quickly gaining in popularity. Major applications including Microsoft Office and Oracle's Internet File System ship with DAV support. Database and application framework vendors are in various stages of offering support for developing DAV views of an underlying relational database. Public domain DAV servers exist that support flat file data repositories and servers based on the open-source MySQL relational databases are expected. Thus, we expect that developing basic DAV interfaces for CMCS community databases will be a relatively straightforward task that can be completed early in the project. NIST databases of thermochemical and kinetics data will be initial targets for this work. It is less clear how quickly support for DASL and query grammars such as XML Query will become available and it may be necessary to implement some basic functionality in these areas to achieve CMCS goals.

While the number of community data resources that will be wrapped as part of the CMCS pilot project is limited, it should be well within the scope of individual researchers and institutions to develop wrappers for additional

databases. The translation and metadata generation capabilities available through SAM should simplify interaction with resources that have already been converted to XML using non-CMCS schema. We will actively pursue such integration activities during the project lifecycle with the aim of building momentum for the long-term support of CMCS capabilities within the community.

Once resources are DAV-enabled, a mechanism is needed to make the portal knowledge-management capabilities aware of them. The registration process should be relatively straight forward, involving the specification of the resource's uniform resource locator (URL) and any schema translations that should be applied when accessing it through the MCS portal. For submission operations, and for non-public resources, interaction of the MCS portal and resource security systems may involve more complexity and require configuration of a GSI-based credential delegation. We will initially develop a form-based resource registration capability accessible through the portal that will support the simple case. We will investigate the practicality of providing web-based configuration for the more general case involving security system integration and work with the CMCS community to determine whether some options, such as the selection of particular schema translations, should be exposed on a per user basis through the general portal customization mechanism. As described in more detail in later sections, we intend to allow personal and group information in notebooks to be registered as searchable resources as well.

1.3.5.4 Multi-scale Integrated Search

The multi-scale search capability within the MCS portal will provide researchers with access to the registered data stores in terms of the community defined interchange schema regardless of the format of the resources themselves. The types of anticipated queries range from simple searches for all measured or calculated values of a molecular property to searches for a contact information for groups who have calculated properties that have significant effects on the uncertainty in a derived reaction rate. The latter, which might be used to initiate a discussion about obtaining updated values of the more fundamental properties, would be a laborious process today, but within CMCS it could be described succinctly and executed automatically. As with databases today, it may be important to provide predefined templates for such complex queries rather than requiring users to formulate them from scratch in the low-level query grammar.

We anticipate that a generic web-based DASL client can provide significant functionality for simple queries since DASL supports methods by which the client can dynamically learn what search grammars are supported and can discover the complete list of property names (keys) that exist. Thus, a generic DASL client may be able to provide scaffolding for helping users construct queries such as drop down lists of the available chemistry-related query terms and search operators, without itself being chemistry specific. We are not aware of such a mature DASL client at this time, so effort may be required within CMCS to enhance a simpler client or to develop one. However, because the client is not chemistry specific, there are significant opportunities to work with SAM and other projects using DASL in its development.

The more complex query example above would rely on the Resource Description Language (RDF) [34] to support semantic relationships and the dependency schema developed in CMCS. The mechanism for this may be derived from the SAM project, which intends to provide an enhanced grammar through DASL to aid in the construction of such queries. Interestingly, since a SAM-enabled notebook is layered and semantic relationships are ultimately defined in XML, the execution of such queries may simply involve server-side translation of the query into the basic DASL grammar within SAM and final execution within the standard DASL engines of federated repositories. Alternatively, similar mechanisms could be developed on the client side within CMCS, although it would involve significant additional work.

1.3.5.5 Content Submission and Retrieval

Once CMCS users have identified information they wish to retrieve, either using the search describe above or pedigree browsing tools described below, or via a reference obtained in another manner, it is a simple matter to download the data to the local machine. If the information needed is metadata, it will already be represented in the XML formatted response to the query, or can be retrieved using the DAV 'propfind' method. If it is data that is required, it can be retrieved using the DAV 'get' method. Since DAV is an extension of HTTP, browsers can execute this 'get' request given the URL of the data item. Capabilities for supporting such requests will be developed in the MCS portal.

While the methods above provide means to retrieve metadata and data, they do not address the issues of making the information usable within applications. To do this, a means of translating the returned values into the data formats expected by the applications is required. Section 1.3.6.3 describes possibilities for implementing such functionality. Similarly, while submission of new data and metadata is conceptually simple and we may provide some simple

capabilities within the portal for manual submission, discussion of the technology to support submission from within applications, and of the policy and procedural issues associated with adding material to a curated data store are delayed until later sections.

1.3.5.6 Data Pedigree and Dependency Browsing

Data pedigree refers to information about how a particular piece of data was produced. In a narrow sense, this implies information about who created the data, when it was produced, and the technique used to create it. In the computational domain, the latter would include information on what software was run, its version number, and the specific parameters used as input to the calculation. As discussed previously, in a broader sense pedigree information may include information on assumptions that limit the data's range of applicability and sensitivity information tracing the uncertainty in a given value to uncertainties in the technique used and uncertainty in values used to calculate it. A pedigree browsing mechanism would allow researchers to discover and navigate through this information.

In CMCS, we anticipate a variety of sources of pedigree information – applications, notebook entries, via metadata generators that extract information from data objects, etc. Such information will be standardized based on the pedigree and dependency schema(s) defined/adopted by the CMCS community. We propose to provide a tool for browsing this information within the MCS portal. Such a tool would provide a visual representation of the pedigree information and allow users to shift focus forward and backward along pedigree chains. SAM specifies a basic component for such navigation. We anticipate guiding its development with requirements from CMCS users and embedding it within a pedigree browsing tool that would be capable of communicating with other MCS tools through drag-and-drop of data URLs, allowing, for example, a user to drag data sets returned in response to a query into the pedigree browser to understand their history.

It should also be noted that the concept of Active Tables can be explained in part as a dependency browsing mechanism. Active Tables combine a means of representing dependencies to the user with additional capabilities to ensure consistency across dependency networks and a means for updating dependency information. Thus, we anticipate opportunities for technology sharing and interactions between Active Tables and the portal pedigree browser that can be explored during the project. Similar overlap exists with the sensitivity analyses involved in chemical mechanism refinement and reduction, which again, may lead to possibilities for linkage with the pedigree browser development.

1.3.6 Research Support

In addition to enabling knowledge management across multi-scale chemical sciences, the CMCS will integrate existing and emerging tools that enhance chemists' efficiency and effectiveness in conducting research, as well as their ability to rapidly share new results with colleagues in multiple disciplines. Capabilities for sharing project documents, developing electronic research records, integrating domain applications with the knowledge management and community capabilities in the MCS portal, and launching applications and distributed computations from within the portal are discussed in the following sections. As with other aspects of the CMCS, the specifics of these capabilities, and the definition of additional ones, will be iteratively refined based on the requirements elicited from CMCS users.

1.3.6.1 Shared Document Repository

Initial discussions with chemists on the CMCS team and our experience in the Diesel Combustion Collaboratory have suggested that a shared document repository allowing papers, proposals, presentations, and figures to be archived in a secure, shared, internet-accessible manner, would provide significant value for working research teams. A number of solutions (e.g., Basic Support for Collaborative Work (BSCW) [35], and Lotus QuickPlace) for a web-based document store exist. Many such products are moving to use DAV as their data storage protocol, which is not surprising given the origins of the DAV standard in this community. CMCS project team members have significant experience with managing shared document repositories and have a good understanding of the features needed to support scientific research groups. Evaluation of the alternatives available and implementation of a basic system will occur early in the project. As CMCS evolves, we will investigate tighter integration of the document repository with other DAV-based aspects of the portal.

1.3.6.2 Electronic Notebook

We expect electronic notebooks to be used heavily by chemical scientists working in the CMCS for a number of reasons. Existing DOE2000 electronic notebooks developed by PNNL and ORNL researchers have gained significant acceptance in the DOE research community. The feature set of current notebooks, including ubiquitous access via the web, the ability for simultaneous use by multiple group members, and rich media support, provides significant value for both individual researchers and distributed teams. As an initial step, we plan to integrate a DOE2000 notebook into the MCS portal. Work currently being performed at PNNL to provide web-based mechanisms for the creating and administering electronic notebooks may make it possible for users to define new notebooks as needed through the MCS portal configuration mechanisms. We also plan to work with the SAM development team to make more powerful DAV- and SAM-enabled electronic notebooks available later in the project. Unlike DOE2000 notebooks, the proposed SAM notebook would expose its metadata and data to applications, agents, and problem-solving environments (PSEs) through the DAV protocol. This provides exciting possibilities for using notebooks to add free-form annotations to data generated by chemistry applications that store data in CMCS repositories using DAV but are otherwise notebook unaware. Conversely, notebook-generated metadata would be automatically available within CMCS search and pedigree browsing tools. The possibilities for ‘publishing’ private annotations from within a notebook to public community metadata repositories is discussed further in section 1.3.7.4. Such capabilities could add significant value in achieving CMCS goals for fostering interdisciplinary communication and automating the flow of information within CMCS. CMCS will provide a testbed for exploring these advanced integration issues, making cutting-edge scientific documentation and records management functionality available to CMCS users and providing invaluable guidance to SAM developers and other notebook integration efforts on maximizing the value of next-generation notebook systems.

1.3.6.3 Portal \leftrightarrow Domain Application Integration

The guiding CMCS scenario posits the use of a variety of domain applications and data resources and bi-directional information flow between them. Analogous to the efforts outlined above to enable access to community databases through a common protocol and using standardized chemical terminology, work will be required to integrate applications into the data and process flows of the CMCS. In many ways, the integration of applications is more challenging than integrating data stores. Applications have no standard interaction protocol, many use proprietary data formats, and they evolve more quickly than databases. For these reasons, integration of the key multi-scale chemistry applications identified in this proposal will be done in an incremental manner and significant application-specific development will be required.

The long-term vision guiding these efforts would be to make applications appear as integral components of the MCS portal. They could be configured to appear as part of researchers customized MCS portal views and would seamlessly exchange data with notebooks and knowledge repositories. Achievement of this vision would evolve the MCS portal into a full-fledged collaborative problem-solving environment (CPSE).

A generic outline of the steps that will be taken to integrate chemistry applications with the CMCS is given here. The details for integrating applications will vary significantly depending on a variety of factors, e.g. whether they have a graphical or command line user interface, whether they use XML, binary, or textual data formats, and whether they store data to flat files, databases, or DAV servers. CMCS developers will work to define detailed implementation plans specific to each application early in the project. As a preliminary step, we will build a web site linking to application documentation, information on obtaining and configuring them, support information, etc. This will be augmented with information about their current state of integration with the CMCS infrastructure, providing a unifying interface to all of the information needed to run the application in the context of CMCS activities. Since the selected applications are already widely used within the community, and their use as stand-alone tools is well understood by CMCS chemists, we will focus initial efforts on metadata and data generation and translation issues. Subsequent efforts will seek to automate the flow of information between applications and CMCS infrastructure.

By design, information available from CMCS that could be considered as input parameters for an application is represented as XML-encoded metadata. (This ignores the trivial case in which a non-XML application input file is stored in a community repository and is re-used by the same application.) Thus, the issue of assembling input information for applications is reduced to the case of taking XML-encoded information and generating an application-specific format. A variety of options exist for performing the required translation. It will be possible to write a custom translator using a language such as Java and standard XML parsing components. A more efficient method may be to use XSL transformations (XSLT) [36] in which the translation is encoded in XML and executed by a standard XSLT engine, eliminating the need for traditional programming. XSL is a stylesheet language for XML. A further option along these lines would be the use of the eXtensible Scientific Interchange Language

(XSIL) [22] and associated Java tools. Similar to using XSLT, this method would allow the required translation to be encoded in XML and executed by a standard engine. However, XSIL's definition of common scientific data structures such as vectors and arrays, and data formatting concepts, such as the binary byte ordering within numerical data types, provide significantly more expressive power. Once a translator has been constructed, a decision will be made whether to integrate it into the MCS data retrieval mechanisms, provide it as a standalone utility, or integrate with the application.

XSIL may also be useful in extracting metadata from the output of applications. The concept of external data streams within XSIL's schema allows the separation of data and the XML description of that data. Thus, XSIL allows a standard XSIL interpreter to combine an XSIL document describing a data format and a file in that format to produce an XML-tagged version of the file from which specific values can easily be extracted. This would be a very powerful technology for generating the metadata needed within CMCS from application output files without traditional programming. An ongoing internal project at PNNL is investigating the limitations of this mechanism in handling complex data files and should be able to provide guidance to the CMCS effort on when the use of XSIL is possible and when development of custom translation software would be required. As with application input translators, a decision will be made whether to integrate translation with the application, with the MCS portal, or to leave it as a standalone utility.

Once translators exist, it should be a relatively small step to enable applications to exchange information via the DAV and DASL protocols. We anticipate the development of Java classes that provide DAV, and a variety of third-party DAV libraries also exist, in a variety of programming languages. Thus it should be possible to develop mechanisms to allow applications to directly query community knowledge bases and submit new chemistry data, chemical metadata, and pedigree information to CMCS. It may also be possible to reuse the user interface components developed for the MCS portal search and pedigree capabilities within applications, providing familiar interfaces for users and reducing development time and effort.

1.3.6.4 Computational Gateway

As the CMCS matures, we will explore the possibility of executing applications from within the MCS portal. This includes both launching the application locally on the user's desktop machine and providing mechanisms to launch remote jobs. In order to implement a computational gateway from the CMCS to applications at distributed sites, we will build on the expertise and technology from the Diesel Combustion Collaboratory (DCC) and grid computing efforts. We will investigate the feasibility of providing remote job launching for the Chemkin application from within the MCS portal earlier in the project through incorporation of the existing web-launch capabilities developed within the DCC. During the CMCS project, we will monitor developments from other SciDAC projects in this area and periodically re-evaluate the opportunities in this area. One area of particular interest would be the development of visual definition of distributed scientific workflows, a capability being proposed within the "Center for Collaborative Problem Solving in the Earth Sciences Community" effort [37]. We envision such a workflow capability could provide a scientific process-oriented view of the multi-scale CMCS tools and resources.

1.3.7 Community Interaction

A third cluster of MCS portal capabilities will provide community interactions. These capabilities are expected to foster general community discussions and aid in community refinement of the CMCS concept itself. Additionally, the community capabilities will provide a bridge between the research/project oriented capabilities and the knowledge management capabilities. Specific capabilities that are planned include a system-wide notification mechanism that can support automated communications and actions triggered by specific events within the CMCS. Community discussion and conferencing capabilities will also be made available. The concept of a community review capability combines these communications mechanisms with tools enabling a process for community acceptance and endorsement of new chemical mechanisms. These capabilities are detailed below. A final section, returning to the issue of access control, attempts to summarize the system-level requirements implied by the combined functionality presented so far, focusing specifically on the migration of data between private, group, and public spaces.

1.3.7.1 Notification

We anticipate notification playing a significant role in fostering cross-disciplinary communication within CMCS. We envision instrumenting much of CMCS to produce notification messages when specific types of events occur. These may include the submission or update of information in the knowledge base, the upload of documents into a project repository, the creation of notes within a notebook, the completion of a computation, new entries within a

discussion tools, or the start and end of videoconferences. Based on such an infrastructure, which will be developed incrementally, we will develop a notification capability for the portal that allows users to specify types of events for which they would like to receive notice. For example, users may request to be notified when new thermochemical data is available about a particular chemical species, or when a chemical mechanism has been updated. We will investigate mechanisms to provide advanced capabilities such as propagating notifications along a dependency relationship such that a fluid dynamicist might receive an automated notice when a new value for a fundamental property of a relevant chemical species might allow an improvement in the model they are using in an engine simulation. We expect significant experimentation will be required to understand what types of notifications produce the highest value for CMCS users.

Notifications will be set up by indicating preferences in a subscription tool in much the same way that one can indicate what sports teams one wants to follow on news-based web portals like Yahoo or Alta-Vista. Subscription information will be stored in user profiles and persist until changed by the user. User feedback will be solicited to determine useful notification types and to prioritize implementation of these capabilities. There are a number of ways that users may want to be “notified”. In most cases, a user would prefer to receive email when a particular event occurs. However, some events may be quite common, and a user may be flooded with email. Another method for notifying users can be displaying a list of event messages at the portal when the user accesses the CMCS. We expect that the notification service will evolve over time as technologies become available that will allow the portal to easily offer user selection of whether notifications are routed to desktop computers, personal digital assistants, pagers, or phones.

We intend to build the notification capability using standard publish/subscribe protocols, e.g. using the Java Messaging Service (JMS) interface. The publish/subscribe model for event propagation is a good fit to our intended use. It maintains the independence of the producers and consumers of notifications, allowing dynamic changes in message routing based on changing consumers subscriptions to various types of published notifications. Additionally, many publish/subscribe systems support persistent messages, which remain cached in the message server until delivered. This is particularly useful in an environment where users connect and disconnect from the system. A variety of public and commercial JMS-compatible message servers exist, and we anticipate the availability of a Grid message service through such a commodity interface in the longer term. This will allow the message server implementation to be selected based on the performance needs of CMCS, and for the implementation to be changed as CMCS grows, without the need for changes to the portal’s notification tool.

1.3.7.2 Chat/Discussion

Tools for including persistent chat and threaded discussion groups as part of a user’s customized portal view will be created. These tools will allow long-running, semi-synchronous conversations between researchers around topics such as model development, community schema definitions, writing papers, etc. We anticipate being able to select an appropriate technology from among existing web-based solutions. Some development work may be required to provide mechanisms for users to discover chat and discussion groups that are available for inclusion in a custom portal view and for users to dynamically create new groups.

1.3.7.3 Conferencing

CMCS will provide a set of video and data conferencing tools to users. In the short term, options such as NetMeeting/SunForum/SGImeting, VNC, WebEx, Lotus Sametime will be considered and we will investigate means to launch these tools from within the portal. Depending on the tools chosen, the size of this effort will vary. In the longer term, we will seek to deploy more powerful and flexible capabilities that may become available from industry, academia, or other DOE/SciDAC projects for example, the Access Grid [38].

1.3.7.4 Community Review

The standard mechanism for evaluating new scientific claims is peer review. Traditionally peer-review has been handled via the scientific literature. However, community data repositories and community systems such as CMCS provide additional means of evaluating research claims and performing peer-review. The powerful capabilities proposed in CMCS to provide data pedigree tracking and to rapidly evaluate the consequences of new chemical information in related sub-disciplines represent a significant advance in the ability of researchers to perform peer-review, both in terms of depth and speed. With the concomitant decreases in the time and effort required to move information from applications and notebooks in CMCS to community knowledge bases, it is clear that CMCS can be a laboratory for the evolution of peer review processes. The new procedures and policies that would enable more efficient and effective peer review given CMCS informatics capabilities are not yet clear. Never-the-less, we propose to develop initial capabilities and policies in this area and evolve it through the life of the project. These

community review capabilities can be tested in the construction of CMCS itself, by using them to reach agreement concerning priorities of CMCS development efforts and the standardization of exchange schema. Bootstrapping the development of acceptable review policies for accepting CMCS data submissions will be another activity that can leverage a prototype capability in this area.

Initially, efforts in this area may be confined to discovering suitable combinations of MCS portal capabilities that can be combined in a ‘community review’ view to support the review process. Mechanisms for voting, scoring, and commenting anonymously will be investigated. Concepts and technologies from research into electronic peer review mechanisms will be investigated with the goal of providing enhanced practical capabilities to CMCS users.

1.3.7.5 Access Management

As with review, CMCS provide significantly new possibilities for automating the migration of data between private, group, and public repositories, yet many of the details of how to harness those possibilities to provide scientific value are unclear. The use of common credentials to provide authentication and authorization control across CMCS, coupled with the capability to annotate information from all sources – applications, notebooks, chat windows, etc. – with metadata specifying which project it relates to would allow the interesting possibility of providing access control based on projects rather than resources. Once again, CMCS can be a laboratory for investigation of new ways of conducting research.

At a practical level, achieving the goals outlined in the CMCS scenario will require the development of effective and practical policies within CMCS for migrating data from private collections through limited access peer-review processes to public databases and back into private areas as input for new research. Based on experience within the DCC and other DOE2000 collaboratories, we can predict the basic access control issues that CMCS will face and be confident that they can be managed, but the wider range of data flows proposed in CMCS present new challenges along with new opportunities and innovation in managing access controls will be important in maximizing the value of CMCS to the chemistry community.

1.3.8 Applications and Data

While the ultimate goal is to provide sustainable capabilities with value to the broad combustion community, the project scope includes funding for a group of chemical science collaborators who are also separately funded to develop tools and/or perform publishable combustion research that will be able to directly use CMCS capabilities. This serves two goals: providing ongoing feedback during CMCS development, and demonstrating proof-of-concept and proof-of-value to the larger community. Similar arrangements, where the dual goals of participatory design and immersive introduction of capabilities into a community are clearly articulated to all project participants, have proved extremely successful in DOE2000 Collaboratory efforts [16,20].

The following subsections describe the tasks to be performed by this ‘chemical science team,’ the community resources that will be accessed, the interactions anticipated with researchers at other scales, and a preliminary analysis of the specific extensions to existing modeling codes and databases necessary to enable these interactions.

1.3.8.1 Community Databases

The chemical science team will be involved in the development of the community standards schema described in section 1.3.5.2. This will involve outreach to their scientific peers to develop schema that will be accepted as sufficient for the broader research community. The initial standards developed will certainly be incomplete, focusing on the cases described below, and will evolve over the course of the project and beyond. Specifically, the initial standards will focus on data that needs to be shared among the specific applications described below. These standards will take advantage of existing efforts, like CML and XSIL, as described earlier. Taking advantage of these developing schema, scientists will incorporate parsers into translators and/or application codes and to enable input and output compatible with these schema.

In addition to the development of the schema, several existing databases will be incorporated into the framework. These include the Kinetics Database at NIST (<http://www.nist.gov/srd/kinet.htm>) and the EMSL Computational Results DataBase (<http://www.emsl.pnl.gov/pub/proj/crdb>) at PNNL. At PNNL this will involve creating translators for specific conversions and possibly some retrofitting of existing databases to new schema. In the initial phases, Ecce will be used as the mechanism to store the molecular data into databases with the appropriate format. As the formats become standard, these will be incorporated into NWChem so that all computations (independent of whether Ecce is used or not) will produce properly formatted data. At NIST the thermodynamic and kinetic databases will be made compatible with DAV and the XML schema. Tools for data submission to these databases

will also be piloted as described in section 1.3.5.3 and 1.3.5.5. NIST will also lead the effort to bring the schema developed in the CMCS in line with evolving standards, in congruence with its mandate to develop standards. It is clear, however, that NIST alone can not support all of the data of interest to the DOE mission. This activity both takes advantage of NIST's leadership and will help understand what additional information the DOE must host.

1.3.8.2 Information Sharing Across the Scales

The scenario described in section 1.3.1 begins at the molecular scale where data for molecules and transition states are calculated using *ab initio* methods as provided by NWChem at PNNL. This data provides input to thermodynamic databases and for transition-state theory methods of calculating reaction rate constants. NWChem will be made accessible either through the MCS Portal or directly through Ecce for interactions with active tables at ANL, and transition-state theory codes associated with the determination or refinement of kinetic rates. Besides the computational databases described above, specific metadata information, such as level of theory, basis sets used, convergence criteria, will be stored in formats which will allow other domain specialists to have access to the data.

Active tables [10,11] use this metadata and the relationships among the thermodynamic properties, also derived from NWChem, to check the consistency of the thermodynamic parameters and their uncertainties as described in sections 1.2.2 and 1.3.1 and Appendix B. Following the active tables analysis, the thermodynamic data will be made available to a larger community of researchers by updating a thermodynamic database. The active table analysis will also indicate where higher fidelity calculations or experiments would be useful. These interactions NWChem will require the development of the appropriate schema, the incorporation of the metadata generation into NWChem data, the communication of that metadata to the active tables, further incorporation of the active tables tool into the portal environment, the means of publishing the results of the active tables analysis to the database. Some of these processes, described in general in section 1.3.5, will initially be manual with the degree of automation increasing over the life of the pilot collaboratory.

Specific challenges exist in the area of active tables metadata. The active tables require metadata describing the uncertainties in the calculations and the relationship between thermodynamic properties. The latter relationships will be calculated through isodesmic reactions in NWChem. A similar set of relationships will be tested in a yet to be determined experimental program. Relational data that transcends a single species (enthalpies of reaction, bond dissociation energies, equilibrium constants) will be stored in a thermochemical network. Both categories of data will be pedigreed via references to source papers. The active tables pilot demonstration will be used to validate the relational schema.

The goal of this activity is to produce an exportable active table technology that will be fully integrated into the MCS Portal environment. In addition to the relational schema development, the CMCS will pilot a collaborative process for publishing relationships to the active table, carrying out the active tables analysis, and documenting output data with appropriate metadata. The ability to publish the results of an active table study to CMCS databases, subject to the review of the database owners, will also be piloted. While the initial effort will concentrate on developing the engine and interfaces using a pilot active thermochemical table, the ultimate goal is to generalize the developed active table technology and expand it toward other active tables (active rates table, active mechanism table, etc.).

The development of chemical mechanisms, described below, also motivates refinement of kinetic rate constants for individual reactions. One means by which this occurs is for NWChem to compute a potential energy surface for a transition state and provide that as input to a transition-state theory code like POLYRATE [39] or VARIFLEX [40] or a more approximate method like QRRK [41]. The CMCS will pilot this interaction between NWChem and at least one of the aforementioned codes. The choice of code(s) will depend on other funding, but team members at ANL, LLNL, MIT, NIST and UCB are familiar with these codes.

Chemical mechanisms are large systems of thermochemical properties about species and the rate constants describing the reactions between those species. A single researcher cannot adequately analyze the many, typically thousands, of parameters involved. The CMCS seeks to facilitate efforts similar to that accomplished in GRI-Mech, where a geographically-dispersed team of researchers collaborated on mechanism development (section 1.2.4). The ultimate goal is to enable this as a part of everyday chemical science research. Taking advantage of kinetics databases like those at NIST and tools for computing kinetic rate constants, scientists at UCB and LLNL will collaboratively develop a chemical mechanism. By bringing researchers into a shared workspace, which the collaboratory would provide, researchers can simultaneously develop the respective regions of a mechanism where they are most knowledgeable, resulting in a mechanism that is the state-of-the-art in a wider range of areas. The base mechanism will likely be GRI-mech (section 1.2.4) with improvements to be made in the area of ignition

chemistry, an area of expertise at LLNL. This will involve the development of schema to describe the kinetic reaction rates and thermodynamic properties and their uncertainties, to describe experimental datasets used for mechanism validation and to describe the performance of the mechanism over the range of experimental datasets. Tools for solution response theory developed at UCB [12,13] and certain Chemkin applications [14] will be modified to work with the schema. The enhancements to the Chemkin package will be pursued in collaboration with Reaction Design (see Consortium Arrangements below). These tools will communicate with the NIST database and the transition-state theory codes described above. They will also take input from experimental datasets used to validate the mechanisms. Information on parameters with both high uncertainty and high sensitivity to performance metrics will be passed to researchers at the smaller scales, motivating additional *ab initio*, active table and transition-state-theory studies. The performance relative to experimental datasets of the resulting mechanism will be documented in the metadata describing the “best-current” mechanism.

The documentation of mechanism performance requires a metric. A simple tool to measure the norm of the error in a prediction relative to an experiment or the difference between two predictions will be developed in this project and applied in a number of the areas. This tool will be used to provide metadata about the ability of models (different levels of *ab initio* approximation, chemical kinetic mechanisms, reduced mechanisms, turbulent combustion submodels) to meet a specified metric, like an experiment or a higher fidelity calculation. This tool will also be useful for documenting model uncertainties where these are not otherwise available.

Since the starting mechanism represents a database, the optimization process involves the modification of information in the data store. This modification is a process that will be studied in CMCS project, including versioning and controlling modification privileges. The chemical mechanism level represents the boundary between molecular scale phenomena and the continuum scales. Tools like the Chemkin applications take mechanisms describing molecular phenomena as input, but describe continuum phenomena like flames.

Chemical kinetic mechanisms comprised of elementary reactions, as described above, are usually computationally expensive for large multidimensional reacting flow simulations. As a result, many means of speeding up the calculation of the source term in these simulations have been developed. These are generally termed reduced mechanisms, though some of them rely on tabulation of changes in the thermochemical state rather than the actual creation of new reduced models. One issue arising is that the reduction process necessarily limits the range of applicability of the mechanism. MIT the CMCS will pilot a means of describing reduced models and their ranges of applicability with XML in a project lead at MIT. The initial focus will be the skeletal models tools developed at MIT, which will be modified to work with schema developed to describe the reduced mechanisms. The process of development is generally the specification of a set of initial conditions for canonical flames for which the mechanism will be developed within a specified error tolerance from a detailed kinetic model. The physical and chemical phenomena (*e.g.* ignition delay times or pollutant formation) that must be modeled to a given fidelity are also specified. The mechanism is validated by comparison of predictive capability relative to the fully detailed kinetic model, assumed to have greater fidelity, over a wider range of initial conditions and canonical flames. Differences between a detailed mechanism and the reduced mechanism are measured as described above, and this information becomes metadata describing the reduced mechanism. In the course of the CMCS project, a variety of mechanism reduction methods will be considered to test the versatility of reduced model XML descriptions. Models that will be considered will include methods of eliminating unimportant reactions from the detailed mechanism [42], methods that develop algebraic steady-state relations [43], methods that rely on tabulation such as PRISM [44]. The sufficiency of the reduced mechanism description schema will be tested through interaction with scientists doing DNS at SNL.

The last stage of the process described in section 1.3.1 is the validation of submodels for turbulent combustion. Two techniques of developing and testing these models are through laboratory experiments and direct numerical simulations (DNS), taking advantage of increases in computing resources. To demonstrate the use of the collaboratory in taking advantage of DNS, we will undertake the mining of DNS data stores developed at Sandia National Laboratories [16] as a part of the BES Chemical Sciences program there. This SNL-led effort will leverage off of a program for feature identification in multidimensional data sets ongoing at SNL; further leveraging off of a proposed SciDAC Chemical Sciences project [45] is also possible. Metadata will be added to the DNS archive to describe the conditions over which the simulations take place. A newly developed tool for identifying and extracting features in time-varying multidimensional datasets, FDTOOLS, will be used to extract features that correspond to phenomena for which turbulent reacting flow models are developed. FDTOOLS will be modified to document these extracted features with appropriate metadata. Based on the metadata associated with the DNS run and the features, reacting flow models will be compared with the extracted features using the comparison tool described in the chemical mechanism section above. These reacting flow models are generally based on certain

canonical flow configurations that can be simulated using the various Chemkin applications, e.g. OppDif or Premix. This will thus involve the execution of Chemkin applications that are initialized from metadata describing the DNS results; eventually the input parameters for the Chemkin applications may be derived from the DNS or FDTOOLS metadata (sections 1.3.6.4 and 1.3.6.5), eliminating some of the human error in creating input files.

This pilot project will test the relationship between metadata descriptions of turbulent and laminar reacting flow data. The initial schema will be sufficient only to describe the current application. However, as with all schemas developed in this project, the collaborative environment will be used to develop more complete schema. An accompanying data store will be developed for archiving the metadata and possibly some of the raw data. This XML schema will be incorporated into Chemkin applications, FDTOOLS and possibly S3D, the DNS code that generated the data discussed in the above paragraph.

The results of the comparison will be metadata that documents the turbulent reacting flow models applicability to the DNS conditions. This will demonstrate the transfer of metadata from a family of modeling codes (FDTOOLS and Chemkin applications) to a description of the turbulent reacting flow model. Eventually, a wide enough range of cases will be tested with the relative error indicated as a function of parameter space. Device scale modelers will be free to take advantage of this information in selecting among the various turbulent reacting flow models that are applicable under different conditions. Errors in predictions will also indicate where model development is desirable. At present, there is little in the way of consistent comparisons between models over a range of parameters.

This completes the flow of information across the scales of relevance to the Chemical Sciences program as described in section 1.3.1.

1.3.9 Outreach to the Scientific User Community

The data interoperability that arises from community standard schema will only be successful if the participation in the schema development by the greater scientific community is sufficiently strong. The application scientists in the CMCS project will conduct outreach seminars and workshops to include the greater scientific community in this development process. The CMCS project as a whole will be benefited by a larger base of collaborators and we will seek to encourage use of the CMCS facility by scientists as it is developed. To this end, the application scientists will also provide demonstrations of the CMCS capabilities at scientific meetings. The CMCS project will also provide workshops on the use of the collaboratory for research groups with outside funding that wish to take advantage of the facility. Application interface components that are developed as a part of the pilot project will be provided as templates for groups wishing to incorporate other application codes into the CMCS framework.

1.3.10 Management Structure

The CMCS project is a three-year project distributed among seven institutions. The development of the infrastructure is centered at SNL and PNL with supporting development at ANL and NIST. The team of application scientists is distributed over the all of the institutions. There are two main simultaneous components to this project. The first component is the development of the MCS portal, CPSE, and data stores that comprise the CMCS infrastructure. The second component is the incorporation of the scientific community and its tools into the CMCS environment.

The construction of a web-based user facility like the CMCS is primarily a systems-integration and engineering task. The CMCS team will iteratively develop capabilities and expand those capabilities over the course of the three-year project. We will endeavor to deploy a version of the CMCS in each year of the project. To do this, we will carry out the following procedures. At the beginning of each year we will plan a series of capabilities to be developed over a half-year period. These capabilities will be tested and refined during the latter half of each year. Before the end of each year, we will demonstrate these capabilities to the scientific community through the each of the scientific applications listed in Sec. 1.3.8 as applicable. We will seek feedback from the scientific community in planning future year activities.

We feel that the measure of success for the CMCS project is the degree to which the scientific community takes advantage of the infrastructure. The team is composed of a number of scientists who seek to improve the collaborative capabilities in the chemical sciences. They will conduct two tasks. They will make common scientific application codes compatible with the CMCS framework. They will also lead the effort among their peers to develop standards of data description that are agreeable to the community; specifically, they will carry out collaborative discussions among their peers to develop standard dictionaries and schema for chemical sciences data. These will be linked to other standards efforts including the NIST program, the w3c group and the CML program

[15] as is feasible. In the beginning, the data standards and code integration will be limited, but as the project progresses the gap between CMCS users will become less evident and geographical barriers will disappear.

To carry out this project, the team will take advantage of the collaborative tools that will be incorporated into the collaboratory. Team meetings will be carried out on a monthly basis with real-time collaboration tools. Subgroups will meet weekly using those same tools. Team discussions will be documented using the asynchronous communication tools described in Sec. 1.3.7. Software development will be carried out using version control software.

The CMCS will employ system administrators for the servers at PNL and SNL as required for day-to-day support of the infrastructure. A server for the NIST databases will be maintained by NIST. All of these institutions have longstanding programs in the chemical sciences as user facilities (PNL and SNL) or providers of standards and databases (NIST). It is hoped that these servers will be deemed valuable and maintained beyond the lifetime of this project through DOE/SC support.

Web software is evolving rapidly. In order to enhance the lifetime of the CMCS infrastructure, we seek to employ technologies that are becoming widely supported. For example, XML and DAV are currently supported by Microsoft, IBM and others.

1.3.11 Timetable

Year 1:

Project Management

- Deploy real-time collaboration tools for CMCS project team. (SNL/PNNL)
- Develop strategy for distributed group conferences for application scientists. (SNL/ANL/PNNL)
- Implement electronic notebooks, web-based archives and discussion groups. (SNL/PNNL)
- Host a project workshop for CMCS project team. (SNL)

MCS Portal (SNL/PNNL/LANL)

- Identify portal-development tools and select portal implementation.
- Build preliminary portal to investigate desired organization of functionality and configuration characteristics desired by users.
- Incorporate threaded discussion and electronic lab notebooks capabilities.
- Provide web-page descriptions of initial data dictionaries for each scale.

Informatics Infrastructure (SNL/PNNL plus organizations listed)

- Identify members for data dictionary/schema definition teams for all scales (members are from collaboratory and interested external parties).
- Investigate current XML data formats and incorporate where applicable. (All)
- Investigate CML and its direction and establish a working relationship. (NIST/ANL/UCB/PNNL)
- Identify tools for parsing, editing, formatting, searching, translating, and processing of XML data. (All)
- Install and configure initial set of data store servers, investigate required security infrastructure, allow sharing of files through these servers. (All)
- Investigate requirements of the notification service, explore implementation options, specify events required by CMCS.

Scientific Applications and Data

- Define initial dictionaries and draft schema for each scale in XML, identify terms that may conflict within a scale, and make data dictionary available from portal. (All)
- Identify tools for parsing and publishing of XML data. Begin development/implementation of these tools into application codes and databases. (All)
- Develop an elementary active thermochemical table using species present in GRI Mech. (ANL)
- Address and analyze version control aspects for active tables. (ANL)
- Establish process to review selected, new thermodynamic and reaction rate constant data added to databases. (LLNL, UC Berkeley)

Year 2:

Project Management

- Enable distributed group conferences for application scientists.
- Conduct second project workshop for CMCS project team with an emphasis on refining draft specifications of data dictionaries and schema.

MCS Portal (SNL/PNNL/LANL)

- Iteratively develop portal and add capabilities including user registration and collection of profile, support for working groups, advertisement of collaborative sessions, and publication of preliminary data dictionaries and schema.
- Provide secure communications to portal.
- Provide initial search capability of metadata-enabled data stores that allows searching based on a selected schema (only data stores that use this schema will be searched).
- Provide capability to retrieve entire data files that are associated with metadata matching the user's search criteria.
- Provide initial data pedigree-browsing capability.

Informatics Infrastructure (SNL/PNNL plus organizations listed)

- Develop tools and/or modify existing tools to parse existing data files, extract data and/or metadata, and publish data and/or metadata to the appropriate data store using the appropriate schema.
- Investigate and deploy appropriate services to allow access to existing NIST reference databases (and possibly at other sites). (NIST/ANL)
- Develop or acquire schema translation tools which provide interoperability of data and metadata across the different scales.
- Provide secure, controlled access to data stores. (ANL)
- Standardize the interfaces for data store access and metadata tools.
- Provide capability to search/query distributed data stores.
- Provide capability to retrieve scientific data linked to specific metadata.
- Prototype search/query access to one or more data stores at more than one scale (multi-schema query).
- Design and pilot simple notification service.
- Finalize definitions of CMCS events.

Scientific Applications and Data

- Refine all data dictionaries and schema into a preliminary data specification. (All)
- Identify the required metadata for the data objects of interest at each scale. (All)
- Demonstration of mechanism reduction documentation with XML. (MIT/LLNL)
- Demonstration of metadata flow from NWChem to active tables and TST codes. (PNNL/ANL/others)
- Test the mechanism development infrastructure for communication between users and database: deposition of new experimental data, review of experimental data, sensitivity analysis, and activation of modeling applications. (UC Berkeley, LLNL)
- Demonstration of FDTOOLS feature documentation. (SNL)
- Implement intelligent decision-making ability to the active table via automated linear analysis. (ANL)
- Develop interactive interfaces to the active table which enable tests of "what-if" scenarios. (ANL)

Year 3:

Project Management

- Integrate advanced electronic notebook capabilities developed.
- Host public workshop with emphasis placed on making the community aware of the tools developed by the CMCS project team.

MCS Portal (SNL/PNNL/LANL)

- Iteratively develop and refine portal capabilities (community communications, improved data pedigree browsing).
- Provide capability to search across all data stores with a base schema used for specifying the query (searches all data stores, regardless of the data store schema).
- Add notification capability to alert users when events of interest (e.g. new data) have occurred.
- Provide execution of scientific application components and codes through portal.
- Investigate and provide prototype capability to extract specific data from stored data files.
- Perform usability study of portal implementation; incorporate results of this study as possible.

Informatics Infrastructure (SNL/PNNL plus organizations listed)

- Refine search/query access to data stores as schema and tools mature.
- Assist in development and integration of metadata publishing tools within the modeling codes and PSEs.
- Provide tighter integration of metadata querying tools with PSEs and applications, allowing more detailed data pedigrees to be completed.
- Fully implement secure CMCS (all data repositories, tools, etc.). (ANL)
- Refine notification service and implement more complicated events.
- Implement XML data formats into codes where appropriate, especially to enable the interoperability of multi-scale data. (All)

Scientific Applications and Data

- Finalize all data dictionary and schema definition standardization activities and present to appropriate standards bodies. (All)
- Merge XML schema arising from sub-disciplines of the chemical sciences to create as comprehensive schema as possible.
- Demonstrate process of optimizing a detailed chemical kinetic mechanism based on new experimental data or new ab initio data. (UC Berkeley, LLNL, PNNL)
- Test implementation of XML data formats into various reduction methods (e.g. reduced mechanism method of algebraic steady-state relations, PRISM). (SNL, UC Berkeley, LLNL)
- Develop interactivity of the active thermochemical tables with the problem-solving environment (Ecce) through common protocols for mining the tables and other data repositories. (ANL)
- Export and generalize the active table technology with the aim of developing rudimentary active rate and mechanism tables. (ANL)

1.4 Subcontractor or Consortium Arrangements

1.4.1 Project Team Consortium

The CMCS project team is spread across five DOE laboratories (SNL, PNNL, ANL, LANL, LLNL), the National Institute of Standards and Technology (NIST) and two universities (MIT and UCB). The team was selected based on member expertise and institutional focus. SNL and PNNL provide the technical focus, drawing on their experience in DOE2000 projects. Researchers from these two DOE laboratories will develop the CMCS infrastructure and much of the portal, and they will organize the knowledge management. ANL provides Grid computing resources, particularly in the area of security. Dave Montoya from LANL is another member of the DOE2000 Diesel Combustion Collaboratory team who will focus on the portal development. NIST is brought into the team because of its role in maintaining public databases of published information in the chemical sciences. In this project Tom Allison from NIST will provide a liaison between the CMCS and (1) the NIST chemical kinetics databases and (2) standards development efforts in the chemical sciences. Dr. Allison is also leading a project at NIST on the development of data standards for chemical kinetics and is an expert in the determination of reaction rate constants.

Scientists at PNNL, ANL, LLNL, SNL, UCB and MIT are drawn from the full range of scales represented in DOE Chemical Sciences research. These scientists will (1) help set user requirements for the CMCS portal, (2) develop acceptable standards for data definitions and schema for data interoperability through interaction with their peers and (3) incorporate pilot scientific application codes and databases into the CMCS infrastructure, providing scientific content for the MCS portal environment. PNNL will implement parts of the Molecular Sciences Software Suite into the CMCS. ANL will implement their active tables tool into the CMCS. ANL will also lead the kinetic reaction rate estimation research area with support from MIT and NIST. Professor Frenklach at UCB is brought to the team for extensive experience in collaborative mechanism development, through the development of GRI-Mech, and will incorporate tools related to mechanism development in the CMCS. Professor Frenklach has also worked with CML and will help develop XML standards and software. MIT will lead the mechanism reduction project area and implement tools developed at MIT for developing and describing reduced chemistry. Professor Green at MIT is also an expert in estimating reaction rate constants, and will provide support for ANL in that area. LLNL is a center for mechanism development among the DOE laboratories and will work with UCB and MIT. SNL is a center for reacting flow research within the DOE laboratories and will incorporate tools related to laminar and turbulent reacting flow modeling into the CMCS.

1.4.2 Reaction Design

The CMCS will work with Reaction Design, a private company that has licensed the Chemkin family of applications. Reaction Design is currently developing XML descriptions (DTDs and schema) for a common self-describing data format, to handle the chemical states that result from reacting flow simulations. The XML development includes incorporation and customization of parser technology, as well as data-compression technology, to handle the large volume of data generated by reacting-flow simulations with 1000s of species and reactions. In addition, they have developed a draft relational database layout that allows storage of relationships between chemistry mechanism sets, individual reaction data, species thermodynamic and transport properties, as well as references and validation history for the stored data. A web-based interface to the database has been prototyped, allowing population and retrieval of data.

1.4.3 Collaborations with other proposed SciDAC projects

The CMCS is a pilot collaboratory. Its objective is to deploy advanced web-based collaborative capabilities to the scientific community. Because the relevant technology is advancing rapidly, we will collaborate with other DOE projects in the area of middleware to bring the most advanced capabilities into the collaboratory. Some of the proposed projects we hope to collaborate with are listed in Appendix D. Our experience with the DOE2000 projects indicates that the dominant technologies also change in the period of several years, the time frame for piloting the CMCS. As such, it is likely that our list of collaborators will evolve as some technologies appear more suitable than others. To bring these advanced collaborative capabilities to the scientific community we will also partner with several proposed SciDAC projects in the Chemical Sciences area. These are also described in Appendix D.

2 Literature Cited/References

- [1] XML Specification, <http://www.w3.org/TR/2000/REC-xml-20001006>.
- [2] J. S. Binkley, *et. al.*, "Combustion Simulation and Modeling," *Proceedings of the Workshop on Combustion Simulation and Modeling*, Reston, VA, June/July 1998.
- [3] C. M. Pancerella, L. A. Rahn, and C. L. Yang, "The Diesel Combustion Collaboratory: Combustion Researchers Collaborating over the Internet", *Proceedings of ACM/IEEE SC99 Conference*, Portland, OR, November 1999.
- [4] R. A. Kendall, E. Aprà, D. E. Bernholdt, E. J. Bylaska, M. Dupuis, G. I. Fann, R. J. Harrison, J. Ju, J. A. Nichols, J. Nieplocha, T. P. Straatsma, T. L. Windus, and A. T. Wong, "High Performance Computational Chemistry; an Overview of NWChem a Distributed Parallel Application," *Computer Physics Communications*, 128, pp. 260, 2000.
- [5] D. E. Bernholdt, E. Aprà, H. A. Früchtl, M.F. Guest, R. J. Harrison, R. A. Kendall, R. A. Kutteh, X. Long, J. B. Nicholas, J. A. Nichols, H. L. Taylor, A. T. Wong, G. I. Fann, R. J. Littlefield, and J. Nieplocha, "Parallel Computational Chemistry Made Easier: The Development of NWChem", *Int. J. Quantum Chem.: Quantum Chem. Symposium 29*, pp. 475-483, 1995.
- [6] M.F. Guest, E. Aprà, D. E. Bernholdt, H. A. Früchtl, R. J. Harrison, R. A. Kendall, R. A. Kutteh, X. Long, J. B. Nicholas, J. A. Nichols, H. L. Taylor, A. T. Wong, G. I. Fann, R. J. Littlefield, and J. Nieplocha, "High Performance Computational Chemistry: NWChem and Fully Distributed Parallel Applications", in *Advances in Parallel Computing*, 10, *High Performance Computing: Technology, Methods, and Applications*, Eds. J. Dongarra, L. Gradinetti, G. Joubert, and J. Kowalik, (Elsevier Science B. V.), pp. 395-427, 1995.
- [7] Extensible Computational Chemistry Environment, <http://www.emsl.pnl.gov:2080/docs/ecce/index.html>
- [8] RFC 2616 Hypertext Transfer Protocol -- HTTP/1.1, <ftp://ftp.isi.edu/in-notes/rfc2616.txt>, June 1999.
- [9] E. J. Whitehead, Jr. and M. Wiggins, "WebDAV: IETF Standard for Collaborative Authoring on the Web," *IEEE Internet Computing*, Vol. 2, No. 5, pp. 34, September-October 1998.
- [10] B. Ruscic, J. V. Michael, P. C. Redfern, L. A. Curtiss, and K. Raghavachari, "Simultaneous Adjustment of Experimentally Based Enthalpies of Formation of CF₃X, X = nil, H, Cl, Br, I, CF₃, CN, and a Probe of G3 Theory," *J. Phys. Chem. A*, 102, pp. 10889-10899, 1998.
- [11] B. Ruscic, M. Litorja, and R. L. Asher, "Ionization Energy of Methylene Revisited: Improved Values for the Enthalpy of Formation of CH₂ and the Bond Energy of CH₃ via Simultaneous Solution of the Local Thermochemical Network", *J. Phys. Chem. A*, 103, pp. 8625-8633, 1999.
- [12] M. Frenklach, "Modeling", in *Combustion Chemistry* (W. C. Gardiner, Jr., Ed.), Springer-Verlag, New York, Chap. 7, pp. 423-453, 1984.
- [13] M. Frenklach, H. Wang, and M. Rabinowitz, "Optimization and Analysis of Large Chemical Kinetic Mechanisms Using the Solution Mapping Method — Combustion of Methane," *J. Prog. Energy Combust. Sci.* 18, pp. 47-73, 1992.
- [14] Chemkin, <http://www.ca.sandia.gov/chemkin/>.
- [15] Chemical Markup Language, <http://www.xml-cml.org/>.
- [16] H. N. Hajm, J. H. Chen, J. F. Grcar, R. Armstrong, C. Kennedy, J. Ray, W. Koegler, A. Lutz, M. Allendorf, D. Klinke, A. McDaniel, N. Nystrom, R. Subramanya, and R. Reddy, "MPP DNS of diesel autoignition", SAND2001-8075, November 2000.
- [17] R. Armstrong, D. Gannon, A. Geist, K. Keahey, S. Kohn, L. McInnes, and S. Parker, "Toward a Common Component Architecture for High Performance Scientific Computing" *Proceedings of 1999 Conference on High Performance Distributed Computing*, Redondo Beach, CA, August 1999.
- [18] M. Thompson, W. Johnston, S. Mudumbai, G. Hoo, and K. Jackson, "Certificate-based Access Control for Widely distributed Resources", *Usenix Security Symposium '99*, March 1999.

- [19] Session Directories for Setting up and Monitoring CORE2000/Habanero Conferences via Java, CORBA, and LDAP, <http://www.emsl.pnl.gov:2080/docs/collab/presentations/papers/wsd.WebNet98.html>.
- [20] J. D. Myers, C. Fox-Dobbs, J. Laird, *et al.*, "Electronic Laboratory Notebooks for Collaborative Research", *Proceedings of the Fifth Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE '96)*, Stanford, CA, June 1996.
- [21] K. A. Keating, J. D. Myers, J. G. Pelton, R. A. Bair, D. E. Wemmer, and P. D. Ellis, "Development and Use of a Virtual NMR Facility", *Journal of Magnetic Resonance*, 143, pp. 172-183, 2000.
- [22] XSIL – Extensible Scientific Interchange Language, <http://www.cacr.caltech.edu/SDA/xsil/>.
- [23] I. Foster and C. Kesselman (eds.), *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Publishers, 1998.
- [24] UMWorktools, <https://worktools.si.umich.edu/>.
- [25] NCSA's OPIE system, <http://www.ncsa.uiuc.edu/opie/>.
- [26] The NPACI User HotPage, <https://hotpage.npaci.edu/>.
- [27] PortalML (Portal Markup Language), <http://www.oasis-open.org/cover/portalML.html>.
- [28] R. Butler, D. Engert, I. Foster, C. Kesselman, S. Tuecke, J. Volmer, and V. Welch, "A National-Scale Authentication Infrastructure", *IEEE Computer*, 33(12), pp. 60-66, 2000.
- [29] S. Tuecke and C. Kesselman, "Security and Policy for Group Collaboration," Proposal to SciDAC 01-06, March 2001.
- [30] M. Thompson, "Distributed Security Architectures: Middleware for Distributed Computing," Proposal to SciDAC 01-06, March 2001.
- [31] "NIST Databases" - http://www.nist.gov/public_affairs/database.htm
 "NIST Chemistry WebBook"- <http://webbook.nist.gov/chemistry/>,
 "NIST 17. NIST Chemical Kinetics Database: Version 2Q98" - <http://www.nist.gov/srd/kinet.htm>
 "NIST Standard Reference Data – Thermochemical Databases" - <http://www.nist.gov/srd/thermo.htm>
 "NIST Standard Reference Data Products Catalog (Surface Data)" - <http://www.nist.gov/srd/surface.htm>
- [32] J. Myers, "Scientific Annotation Middleware (SAM)," Proposal to SciDAC 01-06, March 2001.
- [33] "DAV Searching and Locating (DASL) protocol," <http://www.webdav.org/dasl/>.
- [34] "Resource Description Framework (RDF) language," <http://www.w3.org/RDF/>.
- [35] W. Appelt, "WWW-Based Collaboration with the BSCW System," *Proceedings of SOFSEM'99*, Springer Lecture Notes in Computer Science 1725, pp. 66-78, Milvoy (Czech Republic), 1999.
- [36] XSL Transformations (XSLT), <http://www.w3.org/TR/xslt>.
- [37] D. Gracio, "Center for Collaborative Problem Solving in the Earth Sciences," Proposal to SciDAC 01-06, March, 2001.
- [38] Access Grid, <http://www-fp.mcs.anl.gov/fl/accessgrid/default.htm>.
- [39] J. C. Corchado, Y.-Y. Chuang, P. L. Fast, J. Villa, W.-P. Hu, Y.-P. Liu, G. C. Lynch, K. A. Nguyen, C. F. Jackels, V. S. Melissas, B. J. Lynch, I. Rossi, E. L. Coitino, A. Fernandez-Ramos, R. Steckler, B. C. Garrett, A. D. Isaacson, and D. G. Truhlar, POLYRATE, version 8.5.1, University of Minnesota, Minneapolis, MN, 2000.
- [40] S. J. Klippenstein, A. F. Wagner, R. C. Dunbar, D. M. Wardlaw, S. Robertson, and J. A. Miller, VARIFLEX, version 1.07, A Chemical Kinetics Computer Program, Argonne National Laboratory, December 2000.
- [41] R. G. Susnow, A. M. Dean, W. H. Green, P. K. Peczak, and L. J. Broadbelt, "Rate-Based Construction of Kinetic Models for Complex Systems", *Journal of Physical Chemistry*, A 101, pp. 3731-40, 1997.
- [42] B. Bhattacharjee, W.H. Green, and P.I. Barton, "Globally Optimal Model Reduction", presented at the AIChE National Meeting, Los Angeles, CA, November 2000.
- [43] J. C. Hewson and M. Bollig, "Reduced Mechanisms for NO_x Emissions from Hydrocarbon Diffusion Flames," *Proc. of The Combustion Institute*, 26, pp. 2171-2179, 1996.

- [44] M. Bollig, H. Pitsch, J. C. Hewson, and K. Seshadri, "Reduced n-Heptane Mechanism for Nonpremixed Combustion," *Proc. of The Combustion Institute*, 26, pp. 729-737, 1996.
- [45] S. R. Tonse, N. W. Moriarity, N. J. Brown, and M. Frenklach, "PRISM: Piecewise Reusable Implementation Strategy for Chemical Kinetics," *Israel J. Chem.*, 39, pp. 97-106, 1999.
- [46] A. Trouve, "Terascale High-Fidelity Simulations of Turbulent Combustion with Detailed Chemistry," submitted to SciDAC 01-08, March 2001.

6 Description of Facilities and Resources

6.1 Sandia National Laboratories

Sandia has several resources particularly pertinent to this collaboratory pilot proposal. A Netscape Certificate Server is installed and available for issuing and authenticating PKI certificates. The Diesel Combustion Collaboratory server currently hosts electronic notebooks, data archives, shared filespaces and application codes. An Access Grid node is also available for group-to-group collaboration over Internet2. Sandia is also the home of the Combustion Research Facility and the Visualization Design Center.

6.2 Pacific Northwest National Laboratory

See Appendix C, Section 7 for a description of Environmental Molecular Sciences Laboratory (EMSL) Computational Facilities and Capabilities.

6.3 Argonne National Laboratory

Argonne MCS facilities include major parallel computing systems, I/O subsystems, visualization subsystems, advanced display environments, collaborative environments and high capacity external network links. A 574-processor Linux cluster, a 128-processor Silicon Graphics Origin 2000/Onyx 2 system with 12 Infinite Reality 2 graphics pipelines, and an 80-node IBM SP serve as the primary computation engines and are supported by hierarchical storage with approximately 2.5 TB of disk and a 60 TB tape robot. The SGI serves as the primary visualization server in addition to its use for large computational science experiments. All subsystems, as well as desktop workstations and various servers, are interconnected at gigabit ethernet speeds.

For high-end visualization, MCS maintains multiple immersive virtual reality devices including a 4-wall CAVE and 4 ImmersaDesks connected to a video-switching infrastructure that allows the display devices to be driven by either the SGI Onyx 2 or the Linux cluster. MCS also has three large-format mega-pixel tiled displays, one with ~11 million pixels, and two compact versions with ~3 million pixels each.

In addition, MCS currently supports four group-to-group collaboration environments (Access Grid nodes). The *Access Grid* is an ensemble of resources that supports group-to-group human interaction across the Internet. It consists of large-format multimedia displays, presentation and interactive software environments, interfaces to middleware, and interfaces to remote visualization environments. The Access Grid promotes group-to-group collaboration and communication for 3 to ~20 people per site with 2 to ~10 sites per session. Large-format displays integrated with intelligent or active meeting rooms are a central feature of Access Grid nodes.

6.4 NIST

The National Institute of Standards and Technology (NIST) has a long-standing tradition of excellence in the field of chemical data. Over the past fifty years they have produced widely used data products such as the NIST-JANAF Thermodynamic Tables, the NIST Chemical Kinetics Database, and the NIST Mass-Spectrometry Database. The NIST Chemical Kinetics Database has been developed at NIST for most than ten years. At the present time there is a new effort underway to augment the literature-abstracted data in the database with reviewed data and data calculated using ab initio quantum chemistry. These efforts involve a large portion of the Experimental Kinetics and Thermodynamics Group.

Recently the Thermodynamics Research Center (TRC) was moved to NIST. This group has extensive experience in the area of thermochemical data. Of particular interest is a well-documented set of standards for the representation of thermodynamics data. The TRC group has expressed a willingness to participate in the development of an XML-based format for thermochemical data.

NIST has a number of facilities which will enable its involvement in the collaboratory. These include the necessary computer resources and personnel to maintain them and video conferencing facilities for meetings.

7.1 Appendix A: Combustion Modeling from the Molecular to Device Scale

Fossil fuel energy supply and fossil-fueled combustion systems are the cornerstones of the industrial and commercial sectors of the U.S. economy, accounting for 85% of the energy consumed in the United States each year. In the private sector the combustion of fossil fuels provide a level of comfort and mobility for U.S. citizens that is unrivaled in the world. Despite continuing investments in alternative energy sources, the importance of hydrocarbon fuels, as they relate to the economy and quality of life in the United States, is unlikely to change in the foreseeable future.

Until very recently, combustion efficiency has not been a major issue. Fossil fuels continue to be inexpensive and the supply of fossil fuels remains stable, although heavy dependence on foreign sources has led to major economic and societal dislocations in the past twenty-five years and may do so again in the future. However, recent changes in international environmental mandates have emerged as strong drivers for increased combustion efficiency. U.S. policy dictates that CO₂ emissions be reduced dramatically during the next ten to fifteen years without adversely impacting the economy. This will have a profound impact on the design and operation of the combustion systems of the future.

The development of predictive computational models for realistic combustion devices is a challenging task. Combustion modeling requires the integration of a broad array of computational physical and chemical models into a computational model of the physical combustion device itself. Combustion systems involve three-dimensional, time-dependent, turbulent flows in complex physical configurations, and, in the case of internal combustion engines or turbines, moving components. Many flows include multiphase effects with liquid droplets and solid particles that are transported through the gas phase. Against this fluid-dynamical backdrop, chemical reactions occur that determine the energy production in the system, as well as the emissions that are produced. For complex fuels, the chemistry involves hundreds to thousands of chemical species participating in thousands of reactions. These chemical reactions occur in an environment that is defined by both thermal conduction and radiation.

In addition to the above, the phenomena that influence combustion span a wide range of spatial and temporal scales. For example, the spatial scale extends from the atomistic scale where chemical reactions occur, 10⁻⁹ meters, through the turbulence scale where the effects of turbulence modulate the chemical reactions, 10⁻⁶-10⁻² meters, up to the scale of the combustion device itself, which is measured in meters. Further, many atomistic processes occur in femtoseconds to nanoseconds (10⁻¹⁵-10⁻⁹ seconds), while the warm-up time for an internal combustion engine is measured in minutes. The computational task of addressing this range of time and length scales is one of the major scientific challenges. The traditional approach is to tackle each regime separately, feeding information from one level into the next higher level, in essence “bootstrapping” up from the atomistic to the device level. One of the major bottlenecks in this approach is the passing of information from one level to the next in a consistent and validated manner.

Chemistry is at the heart of all combustion systems. The set of chemical reactions involved in the combustion of hydrocarbon fuels—the combustion reaction mechanism—is one of the key submodels comprising a complete model of a combustion system. This set of reactions determine the rate at which fuel is consumed and energy is released in the combustion process as well as the conditions under which pollutants will be produced. Thus, a thorough understanding of combustion chemistry is required to reduce unwanted emissions and improve system performance. A large number of chemical species and reactions are involved in the combustion of hydrocarbon fuels. Many of the species have only a fleeting existence, yet are critical for sustaining the combustion process; many of the reactions are only possible in the very high temperatures attained in flames, yet are important in the overall flame chemistry. Although the number of chemical species and reactions that *could* be involved in the combustion of hydrocarbon fuels is enormous, experience has shown that the essential features of the process can be represented by reaction models where the number and types of reactions included in the mechanism is governed by the level of detail required to answer specific combustion questions.

Building a chemical reaction mechanism to model the combustion of a hydrocarbon fuel requires three basic types of chemical data.

- The chemical species and reactions involved in the combustion of the given hydrocarbon fuel.
- The thermochemical properties of the chemical species participating in the combustion process.

- The rates of the chemical reactions involved in the combustion mechanism, including their dependence on temperature and pressure.

Most hydrocarbon fuels involved in practical combustion systems such as automobile engines, industrial boilers and furnaces, and gas turbines are complex mixtures of large hydrocarbon molecules. For example, natural gas contains many other minor species, including chemical additives to enhance performance, in addition to the primary component, which is typically methane. It is a daunting task to obtain all of the above data from laboratory studies—many of the chemical species of interest (*e.g.*, radicals such as OH, HCO, and RO₂) have only a fleeting existence under normal conditions. In addition, the harsh conditions of combustion systems (*e.g.*, temperature in excess of 2,000°C) are difficult to reproduce in the laboratory. Atomistic level calculations such as those enabled by NWChem [Ref NWChem], which have recently attained a new level of accuracy, hold great promise for providing the basic thermochemical and kinetic data needed for developing realistic reaction mechanisms for complex hydrocarbon fuels.

The accuracy of the atomistic-scale calculations can be assessed by comparison with the available experimental data (possibly through the active thermodynamic tables). Chemical reaction information required in device and turbulence-level simulations include thermochemical heats of formation, rates of chemical reactions, and molecular transport properties. A wide range of experimental data is currently available for these quantities in the chemical literature. However, data for some of the species resulting from the combustion of the fuels of primary interest here, methane, are sparse. Furthermore, experiments on these fuels aimed at understanding the role of trace species that could result in harmful emissions have not been pursued extensively in the past.

Obtaining this information for the hundreds of species and thousands of reactions involved in the combustion of complex fuels is a daunting task. These quantities are taken from experimental measurements when such values are available, but when experiments have not been performed or the results are unreliable, theoretical techniques must be used to provide this data. It has only been within the past decade that it has become possible to compute the thermochemical properties of molecules and the rates of chemical reaction to an accuracy suitable for use in comprehensive combustion reaction mechanisms.

In order to compute the thermochemical properties of molecules and the rates of chemical reactions, we must be able to calculate molecular structures, vibrational frequencies, and energies to high accuracy for the molecule or reaction complex. To determine these molecular quantities, one must solve the electronic and nuclear Schrödinger equations.

Over the years a number of approximate methods have been developed to solve the electronic Schrödinger equation, including perturbation theory, coupled cluster, and single- and multireference configuration interaction methods as well as, more recently, quantum Monte Carlo methods and density functional theory. Until very recently, it has not been possible to directly compute the thermodynamic properties of molecules to the accuracy needed in combustion models (*e.g.*, reaction energies to better than 1 kcal/mol, reactions barrier accurate to a few tenths of a kcal/mol). To correct for the errors in the computed enthalpies of formation, a number of semiempirical schemes, such as isodesmic reactions, BAC-MP4, the Gn methods, the CBS method, and SEC and SAC/PCI-X were put forward.

To complete the thermodynamic information, we must find the solution of the nuclear Schrödinger equation for the bound vibrational-rotational states of molecules which require the derivatives of the electronic potential energy. Modern geometry optimization techniques use the gradient to locate the equilibrium geometry of a molecule and harmonic vibrational frequencies can be computed from the Hessian matrix evaluated at the equilibrium geometry. Perturbation theory and other approaches can be used to obtain vibrational anharmonicities using the higher-order derivatives, and anharmonic corrections to the harmonic vibrational frequencies can be important when accuracies on the order of a tenth of a kcal/mol are required. Given the electronic energies, equilibrium geometries, and vibrational frequencies, it is then possible to calculate a variety of thermochemical properties including heats of formation, heat capacities, and entropies. The latter two properties can be calculated using statistical mechanical methods with the largest errors usually arising from the use of the harmonic oscillator approximation and the presence of coupled low frequency modes such as internal rotors.

Combustion reaction kinetics encompasses both pressure-independent and pressure-dependent rate constants. Examples of the former might be a simple abstraction reaction or an addition-elimination reaction at high pressures. Examples of the latter might result from an addition reaction that proceeds through the formation of an intermediate complex or soot condensation. Pressure dependencies also arise in reactions when there is competition between the inelastic relaxation and unimolecular dissociation of a chemically activated species. This means that reaction kinetics as used in combustion actually requires a description of both reactive and inelastic scattering.

Calculation of a pressure-dependent rate constant typically requires three kinds of information. First, there is the cumulative reaction probability (CRP) that contains the kinetic import of the transition state. The CRP is formally the sum of the reaction probabilities connecting all states of the reactants and products accessible at a given energy (E) and total angular momentum (J). That sum is essentially controlled by the local properties of the variationally determined transition state along a given reaction path. Second, there is the density of states that expresses the kinetic information carried by the reactants. For simple abstraction reactions, this information takes the form of the partition function. For reactions that proceed through chemically activated complexes, the density of states of the complex for energies at or above dissociation or isomerization is required. Third, there are the transition probabilities that provide the kinetic import of third body collisions responsible for the pressure dependence of reactions. These transition probabilities describe the inelastic processes that compete with unimolecular decay described by the ratio of the CRP and the density of states. That competition is described by a Master Equation approach. In the absence of pressure effects, the ratio of the CRP and the density of states (expressed as partition functions) leads to the required pressure-independent bimolecular rate constants.

All of this data, both thermodynamic and kinetic, may be either directly or indirectly calculated within an electronic structure code NWChem. Several of the kinetic calculations require additional software such as POLYRATE, which take the electronic structure information as input and output the reaction rate for the system of interest. One of the important activities to be accomplished will be to define an extensible representation of the data (including metadata) and implement that representation within the computational software. The data representation (potentially based on CML) will be defined in cooperation of the domain scientists and the computer scientists.

7.2 Appendix B: Active Thermochemical Tables

All well known thermodynamic tables, whether available in hard copy form or on the web, are a “static” listing of results and error bars. Beneath the listings is a complicated but knowable set of relationships between measurements and calculations of more fundamental quantities whose combination leads to the table entry. The table entries themselves are generally not directly measurable quantities. We want to develop an “active” thermodynamic table in which the relationships are exposed to table users and data evaluators. This has at least three advantages.

- (1) New measurements or calculations of the more fundamental quantities can be propagated instantaneously through out the table. If the archival table keeper is an evaluation panel, this becomes a rapid way of publishing the latest information. Provisional tables of properties not yet approved by evaluators become a rapid way to expose tentative new information to the community. Supporting documents ranging from papers and graphs of raw data to notes that can be attached to the table data further accelerate information exchange.
- (2) The active tables produce answers that are always consistent in a global way with all the data available to the table. This is a marked difference from conventional tables that do not have error bars (and sometimes values as well) that are consistent with a global view of relationships. This is a consequence of the fact that conventional table entries are typically generated from relationships with more fundamental quantities that are examined from an aspect that is highly local to the entry. By the linear analysis feature of active tables (described later) inconsistency in error bars can be highlighted. This is particularly useful when using the active table to examine and consider a new datum. By acquiring a global character, active tables invest more meaning in the error bars of all entries.
- (3) In the course of experimental or theoretical studies, there are frequently questions of the “what if” type. An active table that would allow users to extract portions of it to run “what if” scenarios and see the more global implications of new values for quantities would increase the efficiency of research.
- (4) Active tables placed at key levels on different scales can exchange information up and down various scales and make a global consistency check. This feature adds a new dimension to the flow of information between scales.

In the rest of this outline we discuss how a single entry in the table is currently generated, how tables are traditionally built up, and where active tables will change these approaches.

B.I. How a single entry in a thermochemical table is generated

An entry in a thermochemical table for one species – such as a page in JANAF – consists of a list of thermochemical functions, such as enthalpy of formation, entropy, heat capacity/enthalpy increments, etc given at various temperatures. This is the actual information sought for by the user and the *raison d’être* for the table. However, the actual list can be relatively trivially generated from a handful of more fundamental information: enthalpy of formation at one selected temperature (usually 0 or 298 K), vibrational frequencies, geometry (or even only moments of inertia) and knowledge of the character (i.e. symmetry and multiplicity) of the ground electronic state and perhaps one or more excited states if they are low enough. This data falls roughly into two categories. One category consists of data that is *species specific*, and – to the 0th approximation - changes in those data do not affect significantly the other entries in the thermochemical table. Data in this category are various molecular properties, such as vibrational frequencies, geometry and/or rotational constants, low lying electronic states, etc. This data is selected directly from experiments and/or calculations. The second category, the enthalpy of formation at one selected temperature, *transcends a single species* since it is linked to other enthalpies in the table. Experimental measurements never produce directly enthalpies of formation. Rather, they obtain quantities such as enthalpies of reaction, bond dissociation energies, etc., which are then used to derive the desired enthalpy of formation. A properly compiled table will reference the papers that provide the sources of both categories of data and some evaluation of why some data values in the literature were selected and others were not.

B.II. How a traditional thermochemical table is built up

Because the enthalpy of formation at some temperature comes from direct measurements of energy differences (e.g., a measured enthalpy of reaction), enthalpies of formation for all the other species involved in the chemical reaction have to be known to produce the enthalpy for the species of interest. The standard way in which tables of enthalpies of formation are developed is by an “aufbau” principle, i.e. a tabulation of enthalpies is developed in incremental steps. One starts from the elements in their standard states (where the enthalpy is set to a standard value) and follows a particular sequence of elements. Each step adds the enthalpy of formation of another species to the table, using experimental measurements that relate its thermochemical properties to those that have been determined in previous steps. Once a step is completed, the enthalpy of that species is considered fixed, and is used in following

steps as a constant. This is the crucial aspect of the traditional approach during which a thermochemical table becomes “static”. Every enthalpy in a traditional thermochemical table has a large number of hidden dependencies on other previous entries in the table. Hence, if somebody were to, for example, find a new value for a fundamental measurement such as the dissociation energy of oxygen, there is no transparent way (other than manually recompiling the whole table from scratch) to see how does that affect the value of other species in the “static” table.

B.III. Groundwork of Active Thermochemical Tables

To generate the desired temperature dependent properties, an electronic active thermochemical table needs to contain the core *species-specific data* (molecular properties mentioned above) and *the enthalpy of formation at one temperature*. We will return in a moment to the issue of how the latter is derived. If the table has the necessary (and well-known) formulas that link core spectroscopic data to thermochemical properties, then the desired temperature-dependent information can be easily generated at the table-users command. Since this is generally not a computing-intensive task, it is a better and significantly more flexible approach than storing the temperature-dependent information in the form of computed properties of fitted polynomials, since it allows for easy update. The core data can be linked to references, web addresses, electronic forms of the papers, raw data graphs, etc. as the researcher who contributes information to the table sees fit to submit.

Rather than hard-wired enthalpies of formation at a selected temperature entered as core data, an electronic active thermochemical table contains the relationships to other table entries that are used in deriving the enthalpy of formation at one temperature. Hence, instead of selected enthalpies of formation, the active thermochemical table has as basic entries the experimental bond dissociation energies and enthalpies of reaction (including reference and/or pdf files of the relevant papers) coupled to expression that lead to enthalpies of formation. These relationships must carry the error information as well.

The set of basic entries - experimental bond dissociation energies and enthalpies of reaction – define the underlying *thermochemical network* of the table. The *nodes (vertices)* of the network are the desired enthalpies of formation, and the *links (edges)* are the various measurements connecting the nodes, i.e. the relational information defining the topology of the network. Once the network properly describes the species of interest is set up, a global analysis can be performed to obtain the best set of enthalpies of formation with the most consistent error bars for the whole table. In a mathematical sense, each experimental measurement (including its error bar) of an enthalpy of reaction or of a bond dissociation energy becomes one linear equation in which the unknowns are the enthalpies of formation of the species involved in the underlying chemical reaction. The coefficients reflect the stoichiometry of the chemical reaction that is described. This system of equations *is* the thermochemical network. Once the system of equations is set up, one seeks the “best” solution to the equations to obtain the current enthalpies of formation. The solution must deal in a statistically correct way with competing experimental determinations. The system of equations relating enthalpies to experiments will be, generally speaking, over-determined, and the “best” solution to the system is obtained by minimizing χ^2 in error-weighted space. This automatically takes into account all competing measurements, if their error bars are reasonable, and produces consistent error bars for the solutions to the nodes, i.e. enthalpies of formation.

Whether the assigned error bar on an individual measurements (or link) is realistic or not has to be determined in a step preceding the least square solution, known also as a linear analysis of the network. In this step, one enumerates and checks all closed loops in the network (they have to sum up to zero within the propagated error bars of all measurements involved in a particular loop). When the sum does not check out, the loop is flagged, since it has at least one measurement that is off or has an unreasonably tight error bar. Common sections between flagged loops identify the “wrong” experimental data and also indicate (from the amount of discrepancy) the size that the error bar should have really been, allowing its adjustment. (Amplifying the error bar is tantamount to lowering the weight factor of the corresponding measurement – which in the extreme is equivalent to eliminating that measurement from consideration.) The consistency checks during the linear analysis also involve comparisons of individual competing measurements with their averages, weighted averages, and trial least-square fit solutions. When all error bars are adjusted during these (iterative) consistency checks, the network is ready for the least squares fit.

If all the measurements forming the network were at one and the same temperature, then the picture would be rather simple. However, to interrelate measurements at different temperatures, one also needs to bring in the other data, labeled above as species-dependent. Hence, there is an additional feedback loop that relates enthalpies of formation from network solutions and the detailed, temperature-dependent tabulation of thermochemical properties. As long as the other data involved (vibrational frequencies, geometry...) is not unreasonable, this additional relationship is generally “weaker” than the relationship within the network itself, and can be easily satisfied by an iterative approach involving successive network solutions.

An electronic active thermochemical table with the above capabilities allows all new information to globally propagate through the table. Error analysis in the propagation effectively runs in both directions. The error bar of the new experiment may shrink error bars in the table. However, the error bars of other experiments in the table might challenge, via the linear analysis discussed above, the error bar assigned to the new experiment.

An electronic active thermochemical table with the above capabilities allows "what if" experiments. This is most useful if the table-user can select what subset of the table he would want to restrict these experiments to.

The underlying network of an electronic active thermochemical table can start small, and keep expanding by bringing in more literature values and/or new experimental measurements. Even a relatively modest network will provide enough intricacies so that one can develop the necessary mathematical apparatus and software needed to set up this approach. In other words it is amenable to a pilot program.

B.IV. The Active Table Architecture

An active table has an engine and is envisioned to have at least three different interfaces. The engine deals with the details of the relationships in the fundamental data that define the internal underlying network of the active table. The engine analyzes the data network in a global way, examining every constituent piece of information for consistency with the rest of the network and provides the optimal solution to the network. One of the three interfaces presents the optimal solutions to the rest of the universe. While this interface has the complete functionality of a conventional table that can be interactively queried, it has at least two additional benefits. A very important feature is that all the solutions that are presented are guaranteed to be mutually consistent both in value and error bar and consistent in value and error bar with all the data residing in the underlying network. As opposed to a static table, the output is dynamic, in the sense that it can be easily updated (if the user wishes so) to include the latest information. The second interface allows the interactivity with the underlying network. This feature allows changing/editing or adding to fundamental data defining the network, with subsequent initiation of a new solution that is now consistent with the latest changes. The third interface allows interactivity with neighboring tables of other types of properties, which can reside on the same scale or may span different scales. It is through this interface that a change of some property on one scale propagates through the other scales, with a concomitant consistency and error bar check. All three of these interfaces will be compliant with the XML schema developed as part of this proposal, and, in fact, the requirements imposed by the active table architecture will help shape the developed standard.

B.V. Relationship to theoretical calculations

While traditional thermodynamic tables are based almost entirely on experimental data, the active table is amenable to intermingling theory and experiment. The calculations can easily provide direct input into the species-specific molecular properties core data. They can also provide relational data such as bond dissociation energies or isogiric/isodesmic reaction enthalpies, provided that there are means to estimate a realistic confidence limit for the calculated result. Entries in the fundamental data can be easily flagged as of their origin, and the table is then able to perform its analysis using a variety of scenarios. One would be to simply take all available information into account, regardless of whether its origin is in an experiment or a calculation. Another one would be to utilize two network overlays: one that is entirely experimental and another that is based entirely on theory. Either can be used independently to produce an output relying entirely on experiment or on theory. However, a simultaneous but separate analysis of both network overlays offers a tremendous advantage. On the one hand, it provides a meaningful basis for developing generalities that will lead to assignment of realistic error bars for theoretical calculations. On the other hand, this approach can help identify suspect experimental measurements that should be reexamined or remeasured, or it can suggest where improvement in theory is necessary. In addition, by analyzing gaps in the networks, the active table can suggest which particular calculations and experiments would be most useful in adding robustness to the existing body of knowledge. The development of realistic-error bars for calculations is an important issue. So far, most calculations do not have a good way of assigning error bars, and even when they do, they are of a "generic" type, associated to a particular method/basis set combination.

B.VI. Relationship to rates

Rates are directly affected by thermochemistry through equilibrium constants used to obtain reverse rates. They also place lower limits to activation energies. Once the thermochemical active table is fully developed, the active table technology can be exported and adapted to generate other active tables, such as an active rate table. In its simplest form, it would use the exact same approach that is developed for active thermochemical tables. In this case, nodes would be the rates, links would be, for example, relative rates. Reverse rates are obtained by interaction with the thermochemistry active table that will provide equilibrium constants. Also, the active thermochemical table will provide other quantities that can place lower limits on activation energies used in rate expressions. Beyond this point

an active rate table becomes somewhat more complex than a thermochemical active table. In general, many rates are determined from an analysis of a complex measurement, where a number of “auxiliary” rates is assumed to be known and fixed. An active table provides an ideal environment that can propagate changes in those “fixed” rates. This can be achieved by resorting to the original measurements and re-analyzing them within the active table environment. If the original measurements are not easily accessible, one can try to use other schemes, possibly involving reverse-engineering to generate a virtual data set from the original set of rates and then decompose it in light of new information as it becomes available. Clearly, accessing the original data will not be a problem within a collaborative effort. In fact, the development of active tables may eventually lead to new standards requiring in future publications the inclusion of relevant details of the original data in an electronically accessible form.

7.3 Appendix C: Open Data Management Solutions for Problem Solving Environments:
Application of Distributed Authoring and Versioning (DAV) to the Extensible
Computational Chemistry Environment

Open Data Management Solutions for Problem Solving Environments:

Application of Distributed Authoring and Versioning (DAV) to the Extensible Computational Chemistry Environment

Karen Schuchardt, James Myers, Eric Stephan

Pacific Northwest National Laboratory

Karen.Schuchardt@pnl.gov Jim.Myers@pnl.gov Eric.Stephan@pnl.gov

Abstract

Next-generation problem solving environments (PSEs) promise significant advances over those available today. They will span scientific disciplines and incorporate collaborative capabilities. They will host feature-detection and other agents, allow data mining and pedigree tracking, and provide access from a wide range of devices. We believe that fundamental changes in PSE architecture are required to realize these and other PSE goals. This paper focuses specifically on issues related to data management and recommends an approach based on open, metadata-driven repositories with loosely defined, dynamic schemas. We discuss the benefits of this approach and describe the redesign of the Extensible Computational Chemistry Environment's (Ecce) data storage architecture to use such a repository, based on the distributed authoring and versioning (DAV) standard. The suitability of DAV for scientific data, the mapping of Ecce schema to DAV, and promising initial results are presented.

1. Introduction

Scientific problem solving environments are complex computing systems that seek to integrate the activities necessary to accomplish high-level domain tasks [18][19]. They may include components for managing scientific workflow, tracking data pedigrees, transforming and filtering data, analyzing and visualizing results, automating feature extraction, and annotated records management. As described by Gallopoulos et al., they also “use the language of the target class of problems, so users can run them without specialized knowledge of the underlying computer hardware or software”[18]. Thus, at the cognitive level, a PSE encodes domain knowledge, and, to varying degrees, enforces or guides users toward best practices. This characteristic is a powerful benefit of PSEs, particularly for novices or occasional users.

Unfortunately, contemporary PSEs tend to embed domain knowledge into the design of persistent data objects and the data store itself, requiring early agreement about best practices, as well as a complete domain ontology. This process results in several undesirable impacts:

- As the scope of a PSE increases, the number of parties that must agree upon best practices and ontology and the resulting data structures becomes untenably large.
- The incorporation of a component into a PSE requires negotiation between the component developer and the PSE framework designers. Creating a component that fits within a PSE framework often makes the component unusable as a stand-alone application or in another PSE.
- As best practices evolve or the PSEs are extended to support users with different goals, the data structures and control flows must change. All components must be changed simultaneously and the existing data structures migrated.

These problems reduce the ease of PSE evolution, create undesirable coupling between components, and introduce up-front delays in creating and extending PSEs. We believe that advances in data storage architectures are required to mitigate these problems and enable next-generation PSEs. Our work is based on a concept for open, metadata-driven repositories whose schema can be dynamically extended and altered without requiring changes to existing PSE components. This work differs in two respects from the use of metadata in systems such as digital libraries and scientific archives. First, we use the repository as the primary PSE persistence mechanism. Second, it is likely and even expected that no individual component of the system, data store included, will need to understand the entire schema. We believe this significantly reduces the coupling between components and between the data store and components, in turn reducing the level of agreement necessary to create and evolve the PSEs. This paper details our concept and an initial implementation of such an architecture within the Pacific Northwest National Laboratory's Extensible Computational Chemistry Environment (Ecce), and briefly discusses some motivating scenarios made possible by this new design.

2. Background

The Pacific Northwest National Laboratory (PNNL) has several ongoing efforts in developing PSEs, collaboratories, and large-scale data management systems. These efforts focus in different scientific and engineering domains and have developed systems tailored for their respective communities with their differing requirements for security, computation, and data scaling. Unfortunately, the different design choices made with respect to the data management components constrain the scope of applicability of otherwise generic components and pose significant barriers to the development of a single unifying architecture with best-of-breed capabilities.

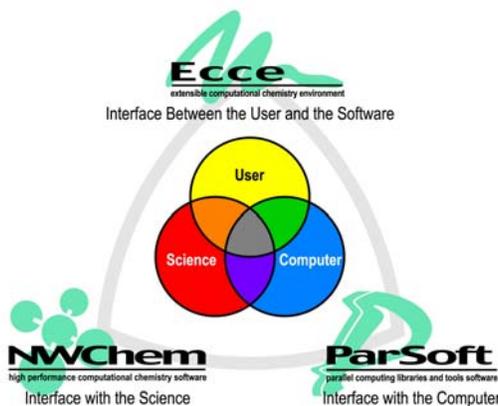


Figure 1. Molecular Science Software Suite (MS3): Ecce, NWChem, and ParSoft

In this paper, the context of PNNL's Ecce is used to explore these issues and to present an initial implementation of an architecture that addresses them. Ecce is one component of the Molecular Science

Software Suite (MS3) - a comprehensive, integrated suite of software that enables scientists to understand complex chemical systems by coupling the power of advanced computational chemistry techniques with existing and rapidly evolving high-performance, massively parallel computing systems. MS3 is represented in Figure 1.

As shown, advanced computational chemistry techniques; ParSoft provides efficient and portable libraries and tools that enable NWChem to run on a wide variety of parallel computing systems. Ecce is a domain-encompassing PSE composed of a set of components to assist the user with many tasks, including the management of projects and calculations, construction of complex molecules and basis sets, generation of input decks, distributed execution of computational models, real-time monitoring, and post-run analysis [3] [4] [13]. Ecce and MS3 have been operational since 1997 and won an R&D 100 award from R&D Magazine in 1999.

Ecce was designed nearly eight years ago around object level integration. At the core of Ecce is an object-oriented chemistry data model that supports the management and manipulation of computational and experimental data and metadata. Until recently, persistent data and the model itself were implemented using an object-oriented database management system (OODBMS). In the early 1990s, object databases were introduced to address nontraditional database application requirements, primarily in engineering and science. Their objective was to combine all of the features of traditional database systems with object-oriented programming languages and eliminate, or at least reduce, the "impedance mismatch" that results from the fundamentally different way that data is represented in databases and programming languages [1] [5].

Ecce designers elected to apply object database technology to the management of complex computational and experimental chemistry data, selecting ObjectStore from Object Design, Inc. (now eXcelon Corporation). Persistent object classes, representing molecules, basis sets, projects, calculations, and jobs, for instance, provided the core around which the suite of tools was developed. These object classes provided the management of data, metadata, and the complex relationships between data objects. This use of object-oriented design and an OODBMS allowed Ecce to provide a high degree of interaction between components, raising the bar for PSE designers by providing a seamless experience for users. At the time of its development, the Ecce architecture represented an innovative approach to addressing the complexity of managing computational chemistry research data [12]. Despite its success, the Ecce design has significant limitations when analyzed in current 2001 terms. OODBM systems have failed to mature and standardize as rapidly as expected. As described in [2], it is nearly impossible to gain complete agreement between vendors on anything concerning object database systems. Although some standards have been defined, they are selectively supported. Other significant problems include proprietary binary formats, tight coupling between the programming language and the OODB, the lack of application development tools, and a painful schema evolution process due to the CODASYL-like schema/application compilation cycle still in use by many products. Object databases have design principles that are opposite from today's dominant *thin-client/fat-server* architecture and cannot leverage the new technologies that have become common in the last several years.

Ecce has first-hand experience with each of these problems. Its OODBMS is built around a memory-mapped client-server architecture with client-side caching. When an object is referenced by a client application, the OODB client software intercepts the request and maps the server-side data into client-side memory. Accomplishing this requires sophisticated client-side libraries and a tight coupling with platform- and vendor-specific compilers – an inherently complex problem with C++ compilers. Because of bugs in the vendor libraries, Ecce has been unable to achieve client-side platform independence, requiring each user to limit their usage to a single platform. Additionally, with the memory-mapped architecture, related objects are mapped to the client at the same time. While this characteristic can be controlled or tuned, it forces applications designers to work with low-level physical layout issues in order to support clients of various capabilities. Modifying the Ecce schema to add new features requires recompilation and distribution of the PSE and migration of persistent data to the new schema. This process is a painful one that encourages design shortcuts to limit the impact of the schema evolution process and that also increases the burden of maintaining an Ecce installation.

Due to lack of standards for OODBMS systems, Ecce has been effectively coupled to a specific OODBMS product whose cost and complexity represent significant barriers to adoption. Finally, the complexity of building, maintaining, and deploying a system based on this technology limits the ability of collaborators to contribute new modules to the PSE. Combined, these issues have made extending, deploying, and supporting Ecce much more challenging than was expected.

Our vision for next generation PSEs is one where independently designed and developed components are rapidly combined to deliver more powerful solutions and reach larger communities of researchers while sharing development costs among the interested parties. For example, Ecce is now adding support for the field of molecular dynamics. This change entails enhancements and additions to the object model and schema that, without changes to the underlying data management system, would have amplified the issues described previously.

Similar problems would be expected during the inclusion of third party tools to compare theoretical and experimental results, to model chemical kinetics, or to add functionality related to biology or materials science. Within PNNL, two existing projects are targeted for integration with Ecce in the near term: a large-scale hierarchical data archive and an Electronic Laboratory Notebook (ELN) system. These systems, which were developed in different languages with different object schemas and data management systems, are essentially third-party applications. Although a useful level of integration has been accomplished with the ELN, the use of independent data stores makes the integration brittle with respect to the evolution of either object model. Direct integration with either of these systems is undesirable due to the resulting tight coupling and impact on deployability and maintainability. Thus, an alternate strategy is required. The reported work to lower development costs and reduce deployment barriers for PSEs, is therefore motivated by practical as well as theoretical considerations: we sought to solve several pressing deployability and integration issues in a manner that would be widely applicable to PSEs in general.

3. Approach

A key observation leading toward an open PSE data management architecture was the realization already noted that PSE components, although they manipulate common data artifacts, often interact through data flow, generating additional attributes or creating new objects related to data generated by another component. This observation leads to several design criteria:

- Direct access to raw data. Access to data through a common object model, although useful in maintaining consistency, limits the representational power of applications added to the system. Providing direct access to the underlying persistent attributes of the data removes this constraint.
- Self-describing data and data relationships. Without an object model common across all applications, another mechanism is needed to allow the discovery of data semantics. Using a self-describing data format (that is, a format that provides metadata about the data), applications can use existing data in new ways and generate new data attributes and relations, as needed. Significantly, applications can also ignore existing relationships that have no meaning for them, or can translate the relationship semantics into their own domain ontology.
- Schema independent data stores. With self-describing data, the data storage system does not need to have deep knowledge of the application objects. By removing knowledge of the schema from the storage system, it becomes possible to support multiple independent or loosely coupled schemas within a single data store where these schemas can evolve without changes to the data store itself.
- Separation of application-level object from the data storage mechanism via a standard protocol(s). Using a standard protocol for describing data management operations helps to maintain the schema-independence previously described. Additionally, a standard protocol allows the selection of the implementation of the data store to be independent of the application technologies. Thus, the data store

can be selected based on the performance, cost, and scaling requirements for a given PSE deployment and on the expected use patterns. Similarly, specifying a protocol instead of a programming interface enables client-side components to be independent of language and platform.

These four bulleted criteria lead to a very flexible, yet powerful, architecture. Applications designed this way can be developed independently, yet integrated deeply based on a partial, post-development mapping between their respective schema descriptions. They can also be deployed to a much broader range of users. Several Ecce-related scenarios, enabled by this design, follow.

Technology Selection

The architecture discussed above could be implemented using a variety of technologies. Data objects that support arbitrary metadata can be developed using the Common Object Request Broker Architecture (CORBA) [21]. Similarly, Version 3 of the Lightweight Directory Access Protocol (LDAP) allows extension of existing entries with new metadata through the use of the extensibleObject class. However, the combination of the Web's Hyper Text Transfer Protocol (HTTP) [8], the Extensible Markup Language (XML) [11], and the Distributed Authoring and Versioning (DAV) protocol, also known as WebDAV [7], provides the closest conceptual mapping to our design goals. DAV, an extension to http 1.1, was originally designed to support collaborative authoring [6]. It provides structured XML-encoded requests for manipulating MIME-typed "documents" (get, put, move, copy, lock) and associated simple text or XML metadata properties (proppatch, propfind). DAV "documents" are not restricted to text-oriented formats and are more analogous to files or binary large objects. New properties can be added at any time, and applications can manipulate arbitrary subsets of properties. For example, an application can request only the values of known properties from the server. Thus, the DAV protocol, with its constructs to logically organize opaque, typed data and to document that data with arbitrary metadata, maps directly into the scientific data management domain.

DAV does support a simple, unordered container/contains relationship, but the wide range of data relationships used in PSEs (for example, temporal, derivative, historical, and sequence, as well as the "is-a" and "has-a" object modeling dependencies) can be encoded using DAV's XML metadata properties. Extensions to DAV, such as DAV Searching and Locating (DASL), Advanced Collections, and Versioning, that are currently under development promise additional PSE-relevant capabilities [9] [15] [16]. XML provides rich capabilities for schema description (XML Schema) and translation (XSLT), avoiding name collisions (XML Namespaces) and representing relationships (Xlink). The emergence of scientific domain languages defined in XML and generic XML parsing tools provide additional leverage. Finally, the maturity of http-related mechanisms for supporting multiple security options and providing scalable performance and fault tolerance, provides a wide range of options for deployment.

Implementation

The discussion of implementing DAV for Ecce includes choosing and testing a DAV server implementation, developing a flexible system architecture, mapping the Ecce data model into the new architecture, and migrating existing data sets.

DAV Server

DAV is quickly gaining popularity in the Web industry. Before the end of 1999, the Apache Software Foundation, IBM, and Microsoft had already deployed DAV servers as extensions to web servers. Client-side support is offered by the Microsoft Office 2000 suite and Java, C++, and Python tool kits. Database vendors are also moving to support DAV. More recently, Web development products have incorporated DAV capabilities including Macromedia Dreamweaver, Adobe Photoshop 6, GoLive 5, and several others. This broad acceptance of DAV is rapidly expanding the server-side options available and the emergence of

optimized, high-performance implementations can be expected. In choosing a DAV server implementation for development use in this project, we emphasized cost, robustness, and protocol conformance over performance. The OpenSource `mod_dav` extension for the Apache Web Server fit these criteria. The `mod_dav` implementation uses file system files and directories to provide persistence for data objects and collections, respectively. Metadata properties are stored in a *hash* table within a database manager (DBM) formatted file, one file per collection. Either simple DBM (SDBM) or Gnu DBM (GDBM) may be used. The Ecce team chose GDBM to avoid SDBM's 1-kilobyte (KB) size limit on individual metadata values. GDBM imposes no size restrictions and has higher performance, although its 25-KB initial size and the lack of automated garbage collection are issues [17]. Because a manual garbage collection utility is available and disk space is relatively inexpensive, these issues were not viewed as critical for our initial implementation.

Under conditions expected to be typical of Ecce requirements, the Ecce team verified the `mod_dav` implementation's DAV protocol compliance, robustness, and performance. Several server configurations were used to assess the effects of key parameters, such as network connection, memory, and operating system features. All servers were built using Apache 1.3.11, `mod_dav` 1.1, and GDBM 1.8. Servers were configured to use basic authentication, to accept persistent connections with limits of 100 connections per minute, 15 seconds between requests, and a minimum of 5 daemons. The test client was a 450-MHz Ultra II with 512 MB RAM. Our client-side software consisted of internally developed C++ classes with 1500-byte packets to mirror our typical TCP packet sizes.

We performed tests to determine upper size limits on documents and metadata. With `mod_dav` and GDBM, it was possible to create and retrieve metadata properties as large as 100 MB, although we currently impose a 10-MB limit for individual properties. There are two reasons for configuring this size restriction. First, storing an XML-based metadata property using `mod_dav` currently requires double the memory of the property: one copy with the XML request body and another copy that is the key/value pair extracted from the body. Second, sending large XML request bodies can create effective denial-of-service attacks. Because initial properties are expected to be in the kilobyte and smaller range, a 10-MB limit does not present a problem. Document size restrictions are those imposed by the underlying file system. Yet, despite testing with property and data sizes much larger than those expected in DAV's prototypical use in document management, we did not find any major DAV protocol compliance issues except for the few noted on the DAV development Web site [10]. These issues did not present any significant problems for the anticipated use.

When the Ecce team looked at DAV as a strategy to replace the Ecce OODBMS, it was unclear if applications built with a request-response protocol (such as http) would provide overall performance comparable with applications built using an OODBMS with a cache-forward architecture. To assess the feasibility, a few tests were developed to mimic typical PSE access patterns. These tests included performing operations on a collection of related objects as a single operation, such as querying for selected properties, copying and removing (Table 1) and moving large data files such as output files (Table 2). All tests were performed during off-hours to minimize the effect of network traffic.

For the operations summarized in Table 1, elapsed and CPU time were collected to help assess whether the time was spent on the client or the server side. Roughly, CPU time is client-side processing time while elapsed time can be interpreted as CPU time plus server and network transport time. Given the relatively small sizes of the metadata and the 150 Mbit/sec ATM network connection, network transport has little impact on these tests thus providing a reasonable assessment of server performance. As shown, metadata operations on individual objects are quite fast. However metadata operations on a large number of objects added up to several seconds. For these operations, more than 50 percent of the time was spent on client-side processing. This percentage can be attributed to the current use of a Document Object Model (DOM)[25] based parser to parse the response and create custom data structures. Significant improvements can be expected by converting to a Simple API for XML (SAX)[26] style parser. In addition, alternative server-side implementations that do not operate on many small metadata databases are expected to provide significant performance improvements. With data distributed across many documents and collections, copy and remove operations can be costly on the server side, but preliminary testing with journaling file systems show that significant performance increases can be expected for these operations as well.

Table 2 shows that our implementation of http/put performed comparably with a standard binary-mode ftp client. It also demonstrates that network bandwidth is the primary driver for moving large amounts of data – our client and server did not introduce bottlenecks.

	Add 1k metadata	Get all metadata Depth=0 ¹	Get metadata Depth=0 ²	Get metadata for selected objects depth=1 ³	Get metadata for selected objects ⁴	Copy hierarchy 50 objects to 4.5MB ⁵	Remove hierarchy with 50 objects totaling 4.5MB
Sun Enterprise 450 ⁷		0.068s	0.055s	2.732s	3.032s	3.482s	1.782s
		0.04s	0.03s	2.04s	1.93s	0.14s	0.01s

Table 1. Performance Results of Typical PSE Operations – elapsed and CPU time

	ftp 20MB mem to mem using /tmp	ftp 20MB local file to local file	ftp 200MB local file to local file	Put 20MB Local file to local	Put 200MB Local file to local
Enterprise 450 ⁸	1.4s	2.4s	23s	3.3s	33s

Table 2. Performance of binary ftp vs http/put

The mod_dav/GDBM/Apache server remained stable during approximately 6 months of testing, using custom test programs running the scenarios above and using the Microsoft Office2000 suite of tools and a Java DAV Explorer client [27] application. During this time, no loss of persistent data or any data transmission loss was experienced.

As of this writing, no performance tests have been run when using alternative authentication mechanisms, such as public key certificates, and no tests of scaling through the use of multi-processor, multi-server load-balancing systems have been done. Because these issues are related to the Apache server, rather than the mod_dav module, the performance hit for secure communications and overall server scalability is expected to be similar to those reported for generic Web applications. Even without performance enhancements such as pipelining and event-based XML parsing, the performance, compatibility, and reliability tests provided confidence that a reliable, deployable system could be built with the current mod_dav implementation.

System Architecture

Figure 2 portrays a high-level view of the system architecture. As shown, although the system uses the Apache/mod_dav server, the system can take advantage of any service that implements the DAV protocol. On the client side is a multi-layered architecture that isolates data access to support plug-in migration and enhancements in the future. Existing Ecce applications can continue to work in terms of its rich set of C++ classes. Factory modules in the object layer encapsulate access to persistent data using implementations of the Data Storage Interface, which maps requests for manipulating data and metadata into protocol-specific

¹ Get all metadata on single document including x system properties and 50 test properties each 1k bytes.

² Query for 5 selected metadata properties on a single document. Same metadata as 1.

³ Use depth=1 capability to query for metadata for 50 objects within a collection.

⁴ Query for selected metadata on 50 objects - one at a time.

⁵ Copy collection of 50 documents each containing 50 1k application properties

⁶ Remove collection created by copy step

⁷ Sun Enterprise 450 running Solaris 2.6 with 512 MB memory and 150 Mbit/sec. ATM network connection. This machine currently serves as Ecce's OODB server.

⁸ Sun Enterprise 450 running Solaris 2.6 with 512 MB memory and 150 Mbit/sec. ATM network connection. This machine currently serves as Ecce's OODB server.

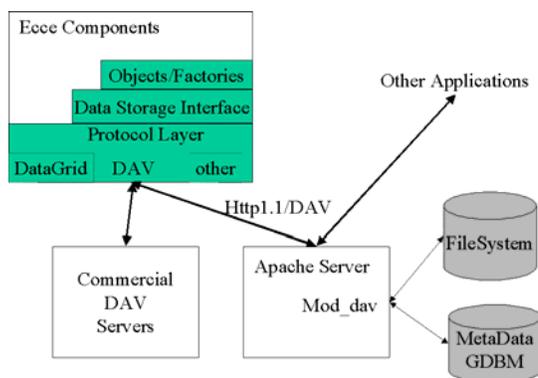


Figure 2. System Architecture Overview

operations. While DAV is the only protocol currently implemented, a separate data storage interface will reduce the changes required to provide native-protocol access to data grids or to incorporate high-performance extensions to DAV – that is, GridDAV analogous to GridFTP[24].

The initial DAV client implementation, based on C++ http classes developed at PNNL and the Apache xerces 1.3 XML DOM parser, is blocking and supports persistent connections, but not pipelining. Further optimization of this implementation, using a SAX parser for example, as well as the extension of the architecture to include a client-side cache, are anticipated. Applications can continue to work at the domain-object abstraction level, but they are also free to access these lower layers, as needed for performance or flexibility, or they may use DAV directly to minimize coupling. As previously noted, the data store is decoupled from Ecce and its only requirement is DAV compliance.

Ecce Schema Mapping

The replacement of the Ecce OODBMS data store with the new architecture has required examination of the use of persistent classes in Ecce and decisions about how to map their structure, content, and relationships into the DAV constructs of collections, documents, and metadata. Ecce had 70 classes “marked” for persistent storage, including relatively simple types, such as dates, and complex class hierarchies that include abstract classes for modeling experiments and calculations, output data properties, molecules, basis sets, and compute jobs. For brevity, the discussion of this mapping process is limited to a subset of the data model – the calculation model. A simplified version of the class model in Unified Modeling Language (UML) notation [20] is shown in Figure 3.

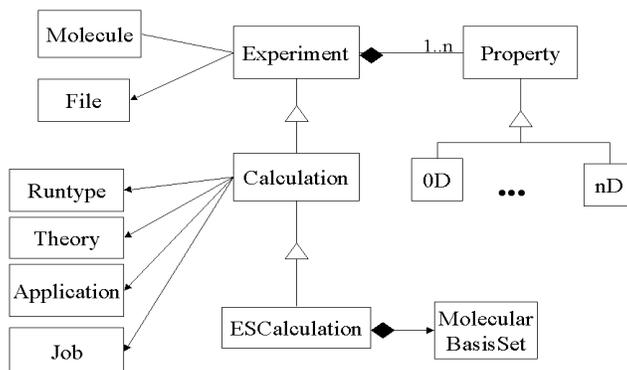


Figure 3 – Simplified Calculation Model

The inheritance in this model provides semantics through virtual methods, as well as through data derivation. Briefly, the model shows a study subject (Molecule) on which an Experiment is performed, the results of which are a series of n-dimensional output Properties. The focus of the model is on simulated experiments or calculations. All the information needed to reproduce the calculation and provide historical context or post-analysis capabilities is captured.

When mapping this model to DAV, the model can be somewhat simplified because the DAV structure does not need to explicitly capture the full inheritance semantics. These semantics can be applied in the object factory layer for applications in which they are important.

Figure 4 depicts how the model was mapped to DAV constructs. In general, objects in the schema were mapped to separate DAV documents. Because DAV supports arbitrary XML-encoded metadata values, it is possible to include related objects within a single document. However, this mapping has several drawbacks. Objects mapped as properties cannot themselves have DAV-accessible metadata properties. Objects mapped as properties also become accessible only through their relationship to the document's main object, severely limiting their ability to participate in multiple relationships.

The team opted instead to map objects to separate DAV documents and in the future will map container relationships through metadata properties. This “*virtual document*” approach increases the granularity of access and assures that all objects are independently accessible and can have their own metadata properties.

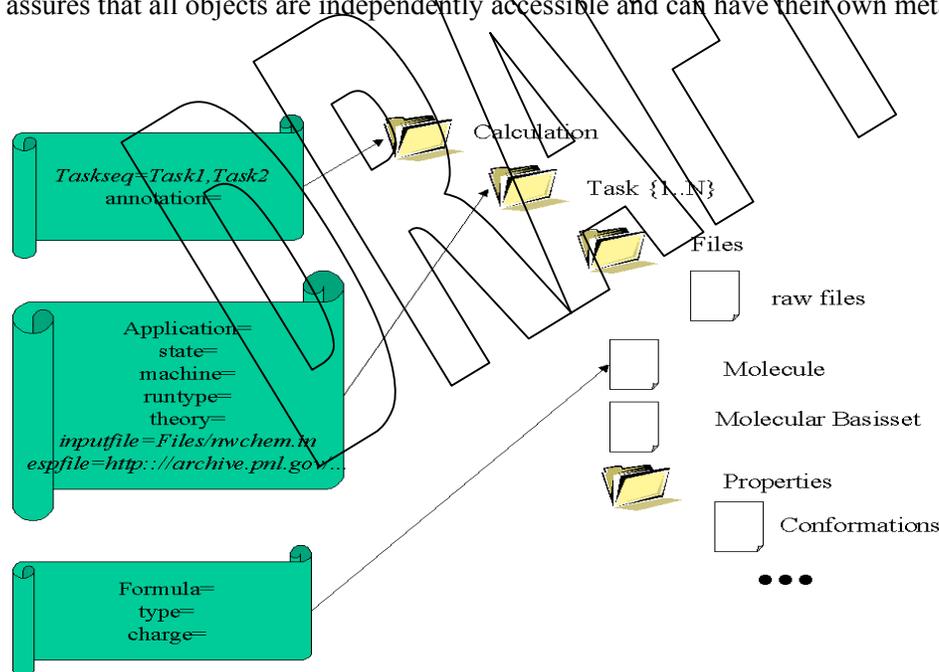


Figure 4. Model Mapped to DAV Constructs

In the initial implementation, hierarchical relationships were mapped into DAV collections. Thus, the list of tasks in a calculation is located through the collection mechanism. This collection-based structuring provides convenience when viewing the data store through standard DAV browsers, but does bend our rule about schema independence. In future implementations, as shown in italics in the diagram, we expect to implement relationships through properties using XML's Xlink semantics and leave all decisions about the use of DAV's collection mechanism to the server. Thus, the physical layout of objects in DAV can be adjusted independent of the metadata-based relationship links. A DAV implementation might elect to store large documents on an

archive system, or perhaps store all documents of a given type, such as 3D molecular structures, in a single hierarchy for easier algorithmic processing. Because the document has self-describing structure, the DAV structure can be reorganized without breaking existing applications, as long as applications interpret the structure dynamically through the metadata.

The data members of individual Ecce objects were mapped to a combination of DAV document data and DAV document metadata properties. Mapping decisions were based on assumptions about other applications that might want to discover, annotate, and manipulate the individual data members. Although these mapping decisions were somewhat arbitrary, the tendency was to decompose Ecce objects as much as possible to increase flexibility, stopping at the point where community standard data structures exist. For example, Ecce's Molecule object was mapped to a Protein Data Bank (PDB) [28] or simple XYZ encoded molecular geometry with metadata properties encoding the empirical formula, symmetry group, and charge state. Thus, applications could search the data store for DAV documents matching the formula metadata and render a 3D display of the molecule without understanding the rest of the Ecce schema. Where standards do not currently exist, plain text or XML markup (where appropriate) is applied to the data, as is done for the Molecular Basis Set document.

For metadata properties, a small number of namespaces were defined to group logically connected properties, rather than develop a single project namespace or XML schema that would be very project specific. As conventions mature and usage becomes widespread, the project will migrate to community standard conventions (for example, the Chemical Markup Language (CML) [22] and naming standards for computational science developed within the GridForum [23]). Table 3 shows a subset of these namespaces and their properties.

Namespace	Example metadata
object	Annotation, citation, reviewed
job	Name, host, directory, nodes, starttime, stoptime, state, maxwalltime, state
molecule	Empiricalformula, charge, symmetrygroup, format
task	Name, runtime, theory, application,

Table 3. Community Standard Conventions: Namespace Subset and Example Properties

Data Migration

Ecce has been operational for a number of years, and existing OODB data sets must be converted to the new storage system. We have conducted preliminary conversions of two of our larger databases, which contain a total of 259 calculations represented by about 420,000 objects with a combined size (excluding raw data files) of 35 MB. We unexpectedly found that the disk requirements increased by about 10%. The bulk of the increase was due to mod_dav: each document or collection may have an associated GDBM database with a default initial size of 25 KB. With our current mapping to DAV collections, a number of these contain much less than 25 KB of metadata. Some of the difference in size can also be attributed to the fact that binary formatted objects such as doubles are typically more compact than textual/XML representations of the same data. While these differences can be explained, we were still surprised because our OODBMS also creates its own overhead using hidden segments to optimize performance. Alternative back-end DAV solutions could be selected to provide more efficient storage, though at current storage costs this is not a pressing concern for Ecce.

4. Discussion

A public beta release of Ecce with the new storage architecture will be available the first quarter of 2001, and design work for adding DAV capabilities to PNNL's electronic notebook and data archive system have begun. As described in following paragraphs, we believe that porting Ecce to the new architecture meets our objectives; it will provide a useful platform for further research and development efforts. A production release of Ecce is expected in June 2001.

At this time, most of the Ecce applications have been converted to the new storage architecture, enabling further performance assessments. Table 4 summarizes size, application startup time, and the operation of each tool loading its set of data for a typical calculation. The selected calculation is of a system with 50 atoms and individual output properties up to 1.8 MB in size. Although enhancing performance was not a primary goal of the project, one primary goal was to avoid a significant performance decrease that would compromise usability. As Table 4 shows, the overall performance actually improved – in some cases significantly. This set of tests is very small; the expectation is that some operations will prove faster with the existing OODBMS version, due in part to its caching mechanism. However, the addition of a cache to the layered client architecture would achieve the same benefits.

Several additional possible optimizations have not been pursued, such as taking advantage of http 1.1 pipelining, making use of multiple simultaneous connections, or bundling requests where class usage patterns involve setting many data members (mapped to metadata on the DAV object) in rapid succession. Note that the test results reported here do not reflect the use of http 1.1 persistent connections. In the current environment, reconnecting each time was significantly faster than making use of persistent connections, an anomaly still under investigation. Overall, these directions, combined with anticipated enhancements in DAV server performance levels, provide a variety of options for substantially improving performance in subsequent Ecce releases.

	Builder	BasisTool	Calc Editor	Calc Viewer	Calc Manager	Job Launcher
7 Ecce 1.5						
Size (res)	30MB	20MB	30MB	30MB	20MB	19M
Cold Start	1.6s	5.0s	2.4s	1.5s	2.8s	0.9s
Warm Start	1.2s	4.6s	2.2s	1.1s	2.7s	0.8s
UO2-15H2O	0.5s	2.14s	7.6s	4.4s	NA	0.95s
8 Ecce 2.0						
Size (res)	25MB	14MB	21MB	25MB	13MB	12MB
Start	1.1s	1.0s	1.0s	0.9s	2.0s	0.42s
UO2-15H2O	0.1s	0.2s	0.9s	2.2s	NA	0.48s

Table 4. Ecce 1.5 vs. Ecce 2.0 alpha Performance Summary
(The client is an Ultra60. Times are elapsed time.)

In terms of deployability, the DAV-enabled Ecce represents a vast improvement. The client and server licensing costs are now zero, assuming use of a no-cost implementation of DAV, such as Apache and mod_dav. Because DAV allows manipulation of individual objects and properties, the memory and processing

requirements on the client are much reduced in comparison with the OODBMS solution. The difference is expected to scale with overall database size. Configuring and running Apache/mod_dav is significantly simpler than installing an OODBMS. Also, because Ecce can share a DAV server with other applications, it is possible to have no server setup at all. This raises the possibility of small academic groups using a departmental DAV server as their data store, or outsourcing the server completely. Although commercial DAV services are aimed more at simple document and file sharing, we have already demonstrated running Ecce against a public DAV server hosted by Xythos [14]. For larger installations, the possibility for using multiple servers with standard web load-balancing and fail-over services (a path not yet explored in detail) promises reliability and scalability. The level of security can also be tailored to group needs; because DAV inherits the HTTP authentication, authorization, and encryption mechanisms, a variety of options exist. The standard HTTP libraries required to support the various web security protocols are not yet installed. When this is done, selecting encryption of communications with the data store becomes a simple matter of Web server configuration. This broad flexibility makes it possible to tailor Ecce to the performance, storage, and management needs of individual groups.

As a testbed, Ecce now provides an unprecedented level of access to its data store, leading to a variety of possibilities. As DAV is an extension of HTTP, Ecce users can run standard Web browsers to “surf” the Ecce database and to view Ecce-generated images, subject to the same access controls applied when accessing the data through Ecce. Existing applets and applications can retrieve and render molecular structures and other data, given the HTTP URL for that item within the Ecce data store.

Developers maintaining and enhancing Ecce have also benefited from the new data architecture. Web and DAV browsers become debugging tools. In-house developers are no longer burdened with a combined application/schema compilation cycle. Third-party developers choose whether to use the Ecce object schema or to develop a mapping of their own objects into DAV using generic XML parsing tools. The latter option will allow electronic notebooks to directly reference and display Ecce data. In addition, the notebooks will have the capability to add additional metadata, such as digital signatures and annotation relationships, to the data without affecting the operation of Ecce. This option also makes possible feature analysis applications or agents that can independently discover objects in the data store (3D structures, for example), apply feature analysis algorithms, and attach their discoveries to the objects as new properties. Although Ecce currently cannot make use of this additional data, we envision a modification that would allow Ecce, or any PSE, to present such metadata to the user as part of a query interface. This generic mechanism would make metadata created by new applications immediately available for use in categorizing and selecting data sets within an existing PSE.

These lightweight integration scenarios can provide real benefits to users without system-wide agreement on a common schema. Moreover, the capability to move incrementally and partially towards a common schema in this open architecture is expected to actually promote more semantic integration. Since DAV supports “live” properties that are calculated dynamically, it is possible to imagine generating metadata on-the-fly to support new applications. Using XML stylesheet language translations (XSLT), a DAV server could be extended to translate properties for applications built using different schema. Thus, developers can encode the mapping between their object schemas external to their applications in a dynamically evolvable form. Although this paper has assumed that such mappings will involve data members encoded in metadata, we are investigating similar mechanisms that would allow XML description of the mapping between the (potentially) binary DAV objects. Ultimately, it may be possible to achieve any desired level of data interoperability between applications through the installation of XML mapping descriptions in a common DAV-based data store.

5. Conclusions

Full realization of this vision will require significant additional work. As noted earlier, many of the advanced features of DAV, including DAV Searching and Locating (DASL) and DAV Access Control, are still being

standardized, while features such as transaction support are not yet addressed. Development tools that simplify extracting metadata from binary data files are also needed, as are mechanisms to dynamically translate between metadata definitions. However, the growing acceptance of XML and DAV should quickly lead to a range of choices in these areas. Once developed, these tools will provide a rich, domain-independent foundation for developing flexible, scalable, evolvable PSEs.

The release of a DAV-based version of Ecce represents a significant advance for current Ecce users and a step towards a more flexible PSE architecture. The development of a new Ecce architecture to use an open, metadata-driven repository based on DAV has provided immediate benefits in terms of flexibility, reduced deployment and maintenance costs, additional security options, and data accessibility. We believe such schema-neutral repositories will be a critical component of next-generation PSE architectures that will enable dynamic collaboration across scientific disciplines and enhance information discovery. Using Ecce as a testbed, the plan is to continue to expand and explore the possibilities inherent in open data architectures for integrating feature detection, data mining, and other agents, along with notebooks and domain applications. Based on early experiences this approach is expected to significantly reduce the barriers to PSE development and evolution while enhancing capabilities and helping to make PSEs a basic part of the scientific infrastructure.

6. Acknowledgments

The Pacific Northwest National Laboratory is operated by Battelle for the U.S. Department of Energy under Contract DE-AC06-76RLO 1830.

7. References

- [1] M. Atkinson, F. Bancillon, D. DeWitt, K. Dittrich, D. Maier, and S. Xdonik. The Object-Oriented Database System Manifesto. In Proceedings of the First International Conference on Deductive and Object-Oriented Databases, pages 223-40, Kyoto, Japan, December 1989.
- [2] Michael J. Carey, David J. DeWitt. Of Objects and Databases: A Decade of Turmoil. Proceedings of the 22nd VLDB Conference Mumbai (Bombay), India, 1996.
- [3] D.A. Dixon, T.H. Dunning, M. Dupuis, D.F. Feller, D.K. Gracio, R.J. Harrison, J.A. Nichols, K.L. Schuchardt. Computational Chemistry in the Environmental Molecular Sciences Laboratory, Plenum Publications, Book Chapter in "High Performance Computing". 1999.
- [4] D.R. Jones, T.L. Keller, K.L. Schuchardt, H.L. Taylor, and D.K. Gracio. Extensible Computation Chemistry Environment (Ecce) Data-Centered Framework for Scientific Research, Wiley Publications, Book Chapter, 1999.
- [5] The Committee For Advanced DBMS Function, Third Generation Database System Manifesto, Computer Standards and Interfaces 13 (1991), pages 41-54. North Holland. Also appears in SIGMOD Record 19:3 Sept, 1990.
- [6] Roy T. Fielding, E. James Whitehead, Jr., Kenneth M. Anderson, Gregory A. Bolcer, Peyman Oreizy, Richard N. Taylor. Web-Based Development of Complex Information Products Communications of the ACM, August 1998 (Vol 41, No 8), pages 84-92.
- [7] RFC 2518 HTTP Extensions for Distributed Authoring -- WEBDAV
- [8] RFC 2616 Hypertext Transfer Protocol -- HTTP/1.1
- [9] DAV Searching & Locating – DASL <http://www.webdav.org/dasl/protocol/draft-dasl-protocol-00.html>
- [10] WebDAV mod_dav http://www.webdav.org/mod_dav/
- [11] XML Specification <http://www.w3.org/TR/2000/REC-xml-20001006>
- [12] External Review Committee Report on the Extensible Computational Chemistry Environment. January 1996.
- [13] The Extensible Computational Chemistry Environment. <http://www.emsl.pnl.gov:2080/docs/Ecce/>
- [14] Xythos. <http://www.xythos.com/>

- [15] WebDAV Ordered Collections Protocol. <http://www.ics.uci.edu/pub/ietf/webdav/collection/draft-ietf-webdav-ordering-protocol-02.txt>
- [16] Goals for Web Versioning. <http://www.webdav.org/deltav/goals/draft-ietf-webdav-version-goals-01.txt>
- [17] DBM Comparisons. http://www.rz.uni-hohenheim.de/anw/prg/perl/nmanual/lib/AnyDBM_File.html
- [18] S. Gallopoulos, E. Houstis, J.R. Rice, "Problem-solving environments for computational Science," IEEE Computational Science and Engineering, Summer, 1994, 11-23
- [19] J.R. Rice & R.F. Boisvert, "From scientific software libraries to problem-solving environments," IEEE Computational Science & Engineering, Fall, 1996, 44-53.
- [20] Unified Modeling Language, <http://www.omg.org/uml>
- [21] PRE - A FRAMEWORK for ENTERPRISE INTEGRATION. R. A. Whiteside, E. J. Friedman-Hill, R. J. Detry. <http://daytona.ca.sandia.gov/pre/s-docs/Information/HICCS.html>
- [22] A universal approach to web-based chemistry using XML and CML. Peter Murray-Rust, Henry S. Rzepa, Michael Write, and Stephan Zara. Chem Commun., 2000, 1471-1472.
- [23] Gridforum. <http://www.gridforum.org>
- [24] GridFTP: Protocol Extensions to FTP for the Grid. W. Allcock, J. Bester, J. Breshnahan, A. Chervenak, L. Liming, S. Tuecke. Internet Draft. March 2001. <http://www.gridforum.org>
- [25] Document Object Model (DOM) Level 2 Core Specification. <http://www.w3.org/TR/DOM-Level-2-Core/>
- [26] Simple API for XML, <http://www.megginson.com/SAX>.
- [27] DAV Explorer, <http://www.ics.uci.edu/~webdav/>
- [28] Protein Data Bank Format, http://www.rcsb.org/pdb/docs/format/pdbguide2.2/guide2.2_frame.html

7.4 Appendix D: Collaboration and Outreach to Other SciDAC Programs

A multi-scale collaboration focused on Chemical Science Discovery through Advanced Computing has been assembled to simultaneously tackle the broad range of issues in Chemical Science. It is envisioned as four coupled sets of activities:

1. "Advanced Methods for Electronic Structure," (submissions to BES/Chem. Sciences) is a collaboration led by PNNL and involving five universities that will provide radical advances in quantum chemical methods to describe efficiently and with controllable precision the electronic structure and dynamics of atoms, molecules and clusters.
2. "Software for the Calculation of Accurate Quantum Mechanical Rate Constants," (submitted to BES/Chemical Sciences) is a collaboration led by ANL involving SNL and three universities that will provide advances in reaction kinetics and dynamics that complement and capitalize on the PNNL advances.
3. "A Computational Facility for Reacting Flow Science," (submissions to BES/Chemical Sciences) is a collaboration led by SNL involving ANL, nine universities, and one industry that will provide next-generation software incorporating new approaches for discovery of chemistry-transport interactions and the predictive description of complex reacting flows that exploit advances in the PNNL and ANL programs.
4. "Collaboratory for Multi-scale Chemical Science," (submitted to OASC/MICS) is a collaboration led by SNL involving ANL, PNNL, LLNL, LANL, NIST, and MIT that will pilot the information and collaboration technology enabling multi-scale knowledge creation, discovery, and exchange within the DOE chemical sciences community.

It is likely that not all of the above proposals will be fully funded. Our intention is to initiate a collaboration that will grow over time into a coordinated approach to the broad range of challenges and opportunities the community faces.

The CMCS will be developed in close collaboration with several other SciDAC projects providing relevant middleware. We anticipate significant benefits in four areas assuming these other projects are funded. First, we seek to employ capabilities to be developed in the "Scientific Annotation Middleware (SAM)" project by PNNL to integrate manual and application-generated annotations into our data stores. Second, we seek to leverage off of further security research proposed in "Distributed Security Architectures: Middleware for Distributed Computing" submitted to SciDAC by LBNL. Third, the "Dynamic Agent Architecture for Collaboration in Context (DAACC)" project at PNNL may facilitate the development of collaborations through data pedigrees and dependencies. Fourth, we seek to take advantage of the development of the Common Component Architecture through the "Center for Component Technology for Terascale Simulation Software" emerging technology center led by SNL to maximize reusability and interchangeability of components developed within this project. We will leverage off of all of these efforts to the extent feasible and beneficial.

There are a number of other proposed projects that would likely provide long-term synergism with the CMCS. Another pilot collaboratory project, the "Center for Collaborative Problem Solving in the Earth Sciences Community," focuses on tools to support the computational workflow that is complementary to the CMCS focus on the data flow. Tools developed in that project may be applied to further develop scientific application codes in the chemical sciences. Work on Grid Computing, the Visualization ETC and the NTON project will develop capabilities for dealing with very large datasets. While these very large datasets are not the focus of information sharing in the CMCS project, a successful project will ultimately need to address expansion in this area.

7.5 Appendix E: EMSL Computational Facilities and Capabilities

From its inception in the early 1990's, the Environmental Molecular Sciences Laboratory (EMSL) has been a major contributor to the evolution in computational chemistry. Confronted with the many extremely challenging fundamental aspects of environmental chemistry, the EMSL established an unprecedented three-prong program in computational chemistry. One effort involved the development of a totally new set of computer codes (the Molecular Sciences Software Suite, MS³) that embody well-established computational models of chemistry to enable modeling and simulations of environmentally relevant chemistry, with models more complex and more accurate than heretofore possible. This software development effort was accomplished by using a new software development paradigm and was focused on taking full advantage of current and future MPP technologies. A second effort involved the creation of a state-of-the-art computational chemistry user-facility, the Molecular Science Computing Facility (MSCF), equipped with a large MPP computer, an experimental MPP system, a data archive system, and a graphics and visualization laboratory. The MSCF is a resource for the chemistry research community with a focus on environmental problems for access to advanced computing facilities and highly efficient computer codes in support of their research. The third effort involves an in-house research program focused on complex chemical problems. Numerous computational studies in support of experimental projects in environmental chemistry as well as new theoretical developments are carried out as part of this program. Significant efforts are also devoted to the further development of new and more accurate chemistry models needed to attack the many complex problems of environmental sciences, with a focus on both electronic structure methods and condensed phase theories.

The Molecular Sciences Computing Facility (MSCF) is located in the William R. Wiley EMSL and is supported by funding from the EMSL project and EMSL operations from OBER. The MSCF contains hardware and software. The hardware consists of a high-performance, massively parallel (MPP) IBM SP supercomputer with 512 processors, 256 gigabytes of memory, and 6 terabytes of disk space with 250 gigaflops of peak performance; a 64 processor next generation SMP parallel computer system from IBM; a graphics and visualization laboratory with the first version of IBM's Scalable Graphics Engine, which connects directly to the switch of an MPP computer; and a scientific database system with 20 terabytes of storage and the Scientific Data Management software. Procurement for an upgrade to the MSCF hardware is underway.

A multidisciplinary team of scientists and computer experts at Pacific Northwest National Laboratory's Environmental Molecular Sciences Laboratory (EMSL) developed MS³. The MS³ consists of three components: 1) the Extensible Computational Chemistry Environment (*Ecce*), 2) the Northwest Computational Chemistry Software (*NWChem*), and 3) Parallel Software Development Tools (*ParSoft*). MS³ won an R&D 100 award in 1999 and a Federal Laboratory Consortium Technology Transfer Award in 2000. MS³ won an R&D 100 award in 1999 and a Federal Laboratory Consortium Technology Transfer Award in 2000.

Ecce is the first comprehensive, integrated, problem-solving environment developed for computational chemistry. Based on an object-oriented data model developed at EMSL, *Ecce* is a suite of distributed client/server applications that enable scientists to easily use computational software such as *NWChem* to perform complex molecular modeling and analysis tasks by accessing networked, high-performance computers from their desktop workstations. *Ecce* combines automated metadata and database management, modern "intelligent" graphical user interfaces, automated calculation initiation and monitoring, scientific visualization, analysis tools, and access to a hierarchical mass storage system. This interactive environment allows the user ready access to computational resources, both hardware and software, on highly sophisticated parallel computing systems. Key components of the *Ecce* environment are:

- Graphical user interfaces for the computational chemistry codes including job setup and launching, job control and monitoring, and computational chemistry advisors, e.g., the Basis Set Browser,
- Visualization software and data analysis for molecular properties.
- Integrated management of the data from molecular computations
- "Data mining" techniques to maximize the use of the computational results.

NWChem^{i,ii,iii,iv,v,vi,vii,viii,ix,x,xi,xii,xiii,xiv,xv,xvi,xvii,xviii,xix,xx} is a new generation of high-performance molecular modeling software that runs on parallel computing systems ranging from clusters of workstations to the emerging teraflops class of massively parallel computers. *NWChem* is scalable to both problem size and computer size as well as portable for different high-performance computing systems. It provides a broad range of capabilities for solving sophisticated mathematical models of chemical systems from first principles at both the molecular orbital and

density functional theory levels. These capabilities enable theoretical chemists to predict the fundamental characteristics of chemical systems at a level of accuracy that is otherwise obtainable only from the most sophisticated experimental approaches. *NWChem* also supports molecular dynamics calculations with a variety of empirical force fields to simulate macromolecular and solution systems as well as with quantum mechanical force fields. The software is modular, so that even though it has more than 500,000 lines of code, less than 10,000 lines must be modified to run at high-performance levels on any new parallel computer architecture. The current version of *NWChem* is 4.0, and its capabilities include:

Molecular electronic structure

- Energies, analytic gradients, and numerical second derivatives by finite difference of the gradients
- ❑ Self Consistent Field (RHF, UHF, high-spin ROHF)
- ❑ Gaussian Density Functional Theory (DFT) with many local and non-local exchange-correlation potentials (RHF and UHF)
- ❑ MP2 including semi-direct using frozen core and RHF or UHF reference
- ❑ Complete active space SCF (CASSCF)
- Energies, numerical gradients and second derivatives by finite difference of the energies
- ❑ MP2, MP4, CCSD(T) with RHF reference
- ❑ MP2 fully-direct with RHF reference
- ❑ MP2 using the Resolution of the Identity integral approximation (RI-MP2)
- ❑ Selected CI with second-order perturbation correction
- ❑ New relativistic treatments including spin-orbit DFT and DKH approach including gradients
- Operations performed by all methods
- ❑ Single point energy
- ❑ Geometry optimization (minimization and transition state)
- ❑ Molecular dynamics on the fully ab initio potential energy surface
- ❑ Normal mode vibrational analysis in cartesian coordinates
- ❑ Generation of an electron density file for graphical display
- ❑ Evaluation of static, one-electron properties
- ❑ Electrostatic potential fit of atomic partial charges
- ❑ Harmonic or hyperbolic RESP restraints
- ❑ Charge group constraints
- ❑ Solvation and environment models (ONIOM and COSMO) are now included for a number of methods
- Interface provided to
- ❑ COLUMBUS multi-reference CI package
- ❑ Natural bond orbital (NBO) package

Periodic system electronic structure

- Gaussian Approach to Polymers, Surfaces and Solids (GAPSS), a DFT based method with many local and non-local exchange-correlation potentials

Classical mechanics

- Force fields
- ❑ Effective pair potentials (AMBER, CHARMM, GROMOS)
- ❑ First order electronic polarization
- ❑ Self consistent polarization
- ❑ Smooth particle-mesh Ewald (PME) long range correction
- ❑ Distance constraints using SHAKE
- Other features
- ❑ Periodic boundary conditions (rectangular or truncated octahedron)
- ❑ Twin range energies and forces
- ❑ Constant pressure scaling
- ❑ Constant temperature scaling
- Operations
- ❑ Single configuration energies
- ❑ Energy minimization
- ❑ Molecular dynamics simulation
- ❑ Free energy simulation (except with PME)
- ❑ Multiconfiguration thermodynamic integration (MCTI)
- ❑ Multistep thermodynamic perturbation (MSTP)

- ❑ Single or dual topology
- ❑ Double-wide sampling
- ❑ Separation-shifted scaling
 - Combined quantum mechanics and classical mechanics*
- Force field
- ❑ Quantum mechanical gradients
- ❑ Classical effective pair potentials
- Operations
- ❑ Single configuration energies
- ❑ Energy minimization
- ❑ Molecular dynamics simulation

ParSoft provides the high-performance, efficient, and portable computing libraries and tools that enable NWChem to run on a wide variety of parallel computing systems with leading-edge performance and scalability. *ParSoft* is targeted at both common and specific research requirements. The parallel software includes the Global Array toolkit, which provides an efficient and portable “shared-memory” programming interface for distributed-memory computers; the Parallel Eigensolver (PeIGS) Library for solving linear algebra on parallel architectures; and Chem I/O, a parallel input/output library. Efficient, portable tools enable a range of applications including *NWChem* to run on a wide variety of parallel computer systems with leading-edge performance and scalability.

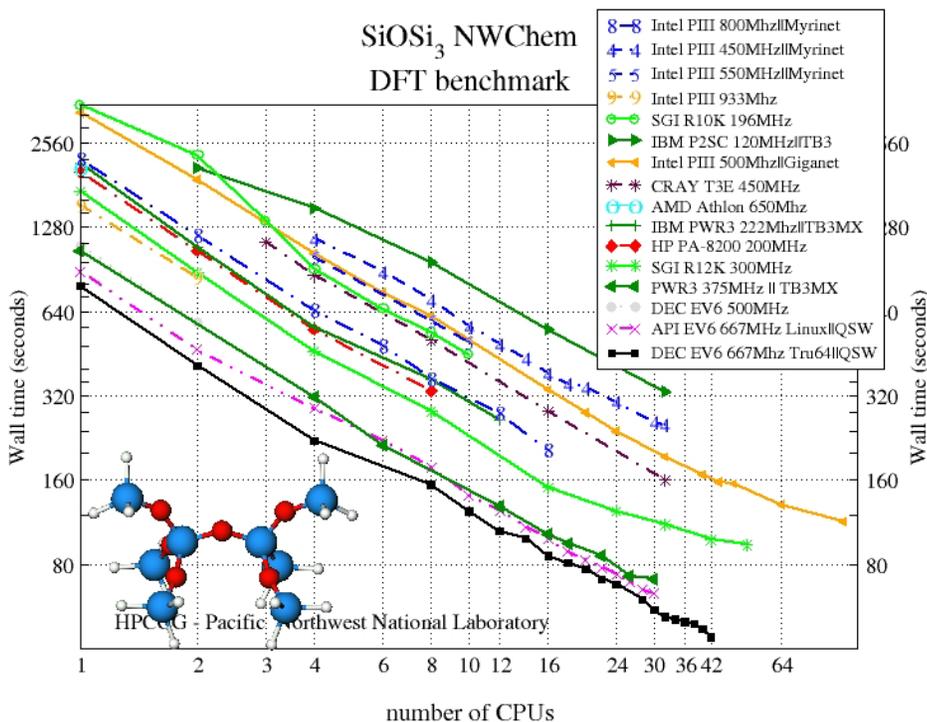


Figure A.1. Time needed to compute the converged local density approximation (LDA) energy of the Si₈O₇H₁₈ zeolite fragment with a basis set of 347 functions.

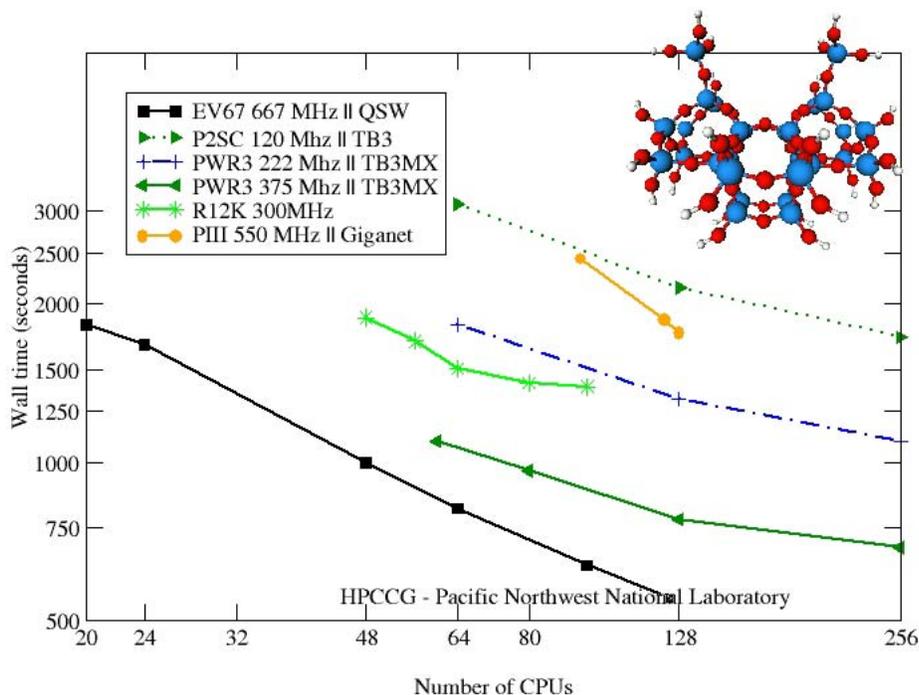


Figure A.2. Time needed to calculate the LDA wavefunction of the $\text{Si}_{28}\text{O}_{67}\text{H}_{30}$ molecule with 1687 basis functions.

MS³ is available to users for no charge through EMSL (<http://www.emsl.pnl.gov>). The *ParSoft* tools can be downloaded and license requests for *Ecce* and *NWChem* can be made from the EMSL web site (<http://www.emsl.pnl.gov>). Additional information on the MS3 components, including how to obtain the software, can be found at the following web pages.

Ecce – <http://www.emsl.pnl.gov:2080/capabs/mscf/software/index.html#ecce>

NWChem – <http://www.emsl.pnl.gov:2080/capabs/mscf/software/index.html#nwchem>

ParSoft – <http://www.emsl.pnl.gov:2080/capabs/mscf/software/hpctools.html>

In addition, other software tools are being developed. In order to optimize the use of the computational resources as well as minimize the user's investment of time, an advisory capability based on prior results and experimental validation is needed. Such a Computational Chemistry Advisor (CCA) can provide information such as what accuracy (e.g. energies or geometries) can be expected from a given basis set and particular treatment of the correlation energy, and the computational cost associated with such a calculation. The CCA is currently under development and will be based on the EMSL Computational Results Database (Feller and Peterson 1998). An example of such an advisor is the Basis Set Advisor that has been developed in the EMSL. As noted above, there are many different forms for the basis sets for treating the 1-particle problem. In order to make a wide range of basis sets, including the correlation-consistent basis sets, generally available, a Basis Set Browser was developed. The Basis Set Browser has made it much easier for a wide range of complex basis sets to be used in many different programs without transcription errors.

Time is available on the four MSCF parallel IBM computer systems for software development, code testing, benchmarking, and some scaling calculations. The group will apply for an MSCF Computational Grand Challenge Allocation under the next allocation call later this Spring.

¹ M. F. Guest, E. Apra, D. E. Bernholdt, H. A. Fruechtl, R. J. Harrison, R. A. Kendall, R. A. Kutteh, X. Long, J. B. Nicholas, J. A. Nichols, H. L. Taylor, A. T. Wong, G. I. Fann, R. J. Littlefield and J. Nieplocha, in *Advances in*

Parallel Computing, **10**, *High Performance Computing: Technology, Methods and Applications*, (Eds.), J. Dongarra, L. Grandinetti, G. Joubert and J. Kowalik, Elsevier Science B.V. p. 395 (1995).

ⁱⁱ M.F. Guest, E. Aprà, D. E. Bernholdt, H. A. Früchtl, R. J. Harrison, R. A. Kendall, R. A. Kutteh, J. B. Nicholas, J. A. Nichols, M. S. Stave, A. T. Wong, R. J. Littlefield and J. Nieplocha, in *High Performance Computing: Symposium 1995*, Grand Challenges in Computer Simulation, Adrian M. Tentner, Editor, Proceedings of the 1995 Simulation Multiconference, April 9-13, 1995, Phoenix, Arizona, Simulation Councils, Inc., The Society for Computer Simulation, San Diego, CA, p. 511 (1995)

ⁱⁱⁱ D. E. Bernholdt, E. Apra, H. Fruechtl, M. F. Guest, R. J. Harrison, R. A. Kendall, R. A. Kutteh, X. Long, J. B. Nicholas, J. Nichols, H. L. Taylor, A. T. Wong, G. I. Fann, R. J. Littlefield, and J. Niepolcha, *Int. J. Quantum Chem: Quantum Chem. Symp.* **29**, 475 (1995).

^{iv} M. F. Guest, E. Aprà, D. E. Bernholdt, H. A. Früchtl, R. J. Harrison, R. A. Kendall, R. A. Kutteh, X. Long, J. B. Nicholas, J. A. Nichols, H. L. Taylor, A. T. Wong, G. I. Fann, R. J. Littlefield and J. Nieplocha, in *Applied Parallel Computing. Computations in Physics, Chemistry, and Engineering Science*, Eds. J. Wasniewski, J. Dongarra, and K. Madsen, *Lecture Notes in Computer Science*, **1041** (Springer-Verlag, Berlin, p. 278, 1996).

^v M. F. Guest, E. Aprà, D. E. Bernholdt, H. A. Früchtl, R. J. Harrison, R. A. Kendall, R. A. Kutteh, X. Long, J. B. Nicholas, J. A. Nichols, H. L. Taylor, A. T. Wong, G. I. Fann, R. J. Littlefield and J. Nieplocha, *Future Generations Computer Systems* **12(4)**, 273, (1996).

^{vi} D. A. Dixon, T. H. Dunning, Jr., M. Dupuis, D. Feller, D. Gracio, R. J. Harrison, D. R. Jones, R. A. Kendall, J. A. Nichols, K. Schuchardt and T. P. Straatsma, *High Performance Computing*, Eds. R. J. Allan et al, (Kluwer Academic, Plenum Publishers, New York. pp. 215, 1999).

^{vii} R. A. Kendall, E. Aprà, D. E. Bernholdt, E. J. Bylaska, M. Dupuis, G. I. Fann, R. J. Harrison, J. Ju, J. A. Nichols, J. Nieplocha, T. P. Straatsma, T. L. Windus, A. T. Wong, *Computer Phys. Comm.* **128**, 260, (2000).

^{viii} T.P.Straatsma, M.Philippopoulos and J.A.McCammon, *Computer Phys. Comm.*, **128**, , (2000).

^{ix} Tilson JL, Minkoff M, Wagner AF, Shepard R, Sutton P, Harrison RJ, Kendall RA and Wong AT. 1999. *Int. J. of High Performance Computing Apps.* **13**, 291 (1999).

^x H.A. Fruchtl, R.A. Kendall, J.A. Nichols, K.G. Dyall and R.J. Harrison, *Int. J. Quantum Chem.* **64**, 63 (1996).

^{xi} T.P.Straatsma and V.Helms, Proceedings Molecular Dynamics on Parallel Computers, 1999.

^{xii} I. T. Foster, J. L. Tilson, A. F. Wagner, R. Shepard, R. J. Harrison, R. A. Kendall, and R. J. Littlefield, *J. Comp. Chem.* **17**, 109 (1996).

^{xiii} R. J. Harrison, M. F. Guest, R. A. Kendall, D. E. Bernholdt, A. T. Wong, M. Stave, J. L. Anchell, A. C. Hess, R. J. Littlefield, G. I. Fann, J. Nieplocha, G. S. Thomas, D. Elwood, J. Tilson, R. L. Shepard, A. F. Wagner, I. T. Foster, E. Lusk and R. Stevens, *J. Comp. Chem.* **17**, 124 (1996).

^{xiv} D. E. Bernholdt and R. J. Harrison, *J. Chem. Phys.* **102**, 9582 (1995).

^{xv} A. T. Wong, R. J. Harrison and A. P. Rendell, *Theo. Chim. Acta* **93**, 317 (1996).

^{xvi} D. E. Bernholdt and R. J. Harrison, *Chem. Phys. Lett.* **250**, 477 (1996).

^{xvii} Dachsel H, Harrison RJ and Dixon DA. 1999. *J. Chemical Physics. J. Phys. Chem. A* **103**, 152 (1999).

^{xviii} J. Garza, J. A. Nichols and D. A. Dixon, *J. Chem. Phys.* **112**, 1150, 7880 (2000); **113**, 6029 (2000).

^{xix} J. Garza, R. Vargas, J. A. Nichols, and D. A. Dixon, *J. Chem. Phys.* **114**, 639 (2001).

^{xx} W.A. de Jong, R.J. Harrison, and D.A. Dixon, *J. Chem. Phys.* **114**, 48 (2001).