PSL Quarterly Progress Report

Project Title: Process Specification and Simulation
Date: June 30, 2001
Principal Investigator: Kincho H. Law, Stanford University
Period Covered: April 1, 2001 – June 30, 2001

Project Objectives

The proposed project is intended to be a feasibility study to evaluate the process specification language (PSL) and a simulation query language (SimQL) with application to project and workflow management. This proposed joint research effort includes:

- to evaluate the potential of the Process Specification Language (PSL) for the planning and modeling activities in a project
- to evaluate the adequacy of the Process Specification Language (PSL) as an interchange definition language to support process-oriented simulation
- to evaluate the applicability of SimQL, a simulation query language, for practical engineering problems
- to develop an integration framework using PSL and SimQL for process-oriented simulation

Our long-term goal is to develop a distributed network-based framework to integrate process specification and modeling and virtual simulations of project activities. To facilitate this research, we use Vite, which is originally developed at Stanford's Center for Integrated Facility Engineering as a benchmarking application for the evaluation of PSL and SimQL.

Progress and Results

Our first goal in this project is to evaluate PSL as process specification interchange standard using Vite as a benchmark application. Vite is a project and organization modeling system designed to assist in developing organizational structures and identifying potential problems with project cost, time, or quality. It takes traditionally qualitative organizational management theory and builds a model that incorporates rough quantitative measures. As for this investigation, we have built a sample demonstration using PSL as an interchange format to exchange information among Primavera's P3, Microsoft Project 2000 and Vite. (Primavera's P3 and Microsoft Project 2000 are project scheduling software widely used in the construction industry.) Presently, the demonstration application includes six translators: Vite to PSL(KIF) translator, PSL(KIF) to P3 translator, P3 to PSL(KIF) translator, PSL(KIF) to Vite translator, Project to PSL(KIF) translator and PSL(KIF) to Project translator. The translation process among P3, Project and Vite using PSL is summarized as shown in Figure 1.

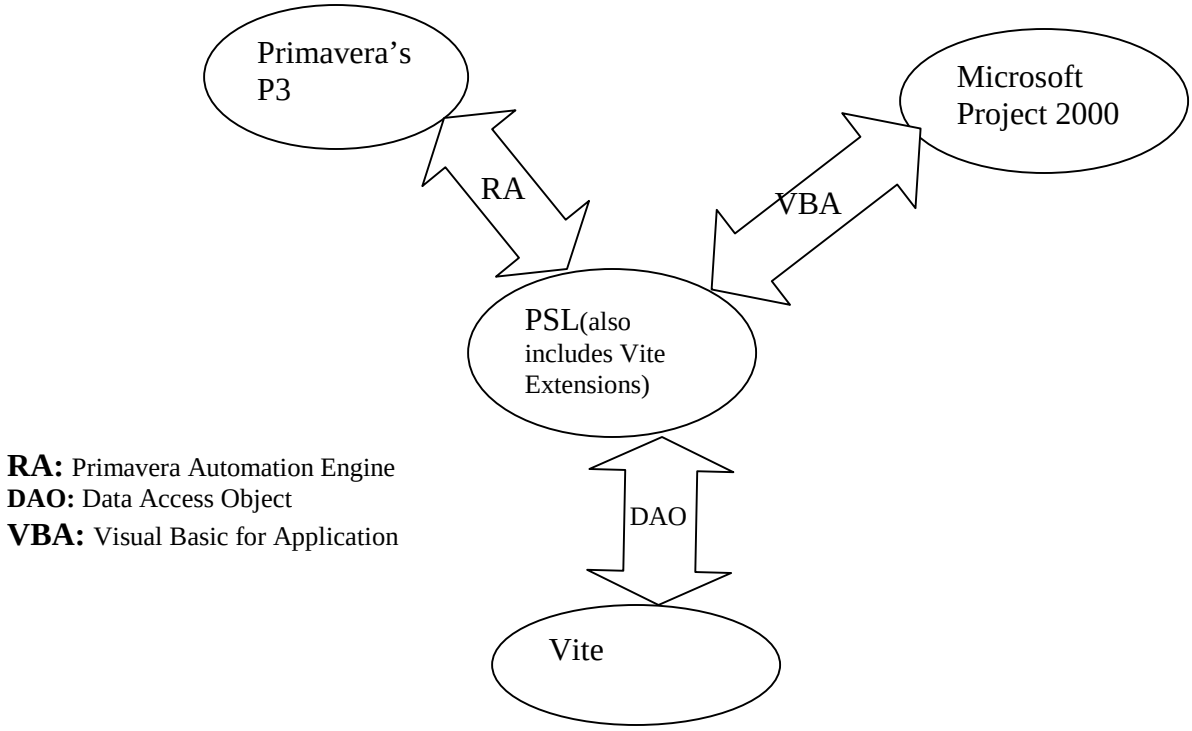

Figure 1: Information Exchange among Vite, P3 and Microsoft Project.

For a construction project, project information includes three basic parts:

- Scheduling
- Resource
- Cost

The relationships among these three parts could be illustrated in figure 2 below:

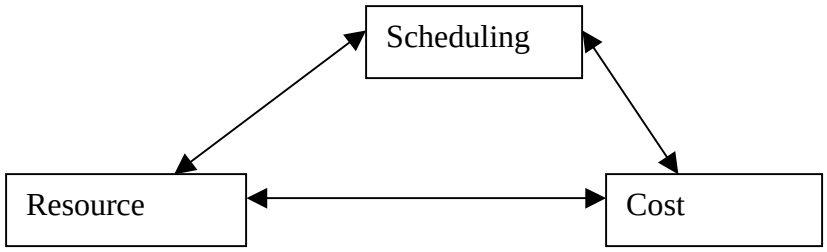

Figure 2: Information in Construction Project

The basic PSL ontology contains primarily the scheduling part. Extensions of the PSL ontology to include resources and costs have been discussed in the previous report. Three basic tasks have been performed during this report period.

- We build sample demonstration programs using PSL to exchange project information among Vite, Primevera's P3 and Microsoft Project 2000.

- We have also investigated on using aecXML (in addition to PSL) to exchange project information among Vite, P3 and Microsoft Project. AecXML is an XML-based language used to represent information in the Architecture, Engineering and Construction (AEC) industry. Here we use aecXML schema provided by Primavera to exchange project information. The aecXML schema proposed by Primavera is relatively complete and contains the essential concepts (scheduling, resource, cost) for a construction project. A comparison report of PSL and aecXML is currently under preparation.

- We have begun an initial investigation on the possible integration of PSL with SimQL

Information Exchange Among Vite, P3 and MS Project

A Vite project is composed of a traditional CPM diagram, additional links showing failure dependence and reciprocal information dependence, a management structure diagram, and responsibility links between the management structure and activities. In this example scenario, we focus on interchanging activity information among Vite , P3 and Microsoft Project. A typical CPM activity diagram is shown in Figure 3. Activities are named and given durations, and groups of activities are linked together using relationships such as finish-start, start-start, or finish-finish. Milestones may be included in the activity diagram, such as "ship tapes to foundry" in the example project. Begin and end milestones are used to denote the start and finish of the project ("Start Project" and "Fab, Test and Deliver.")

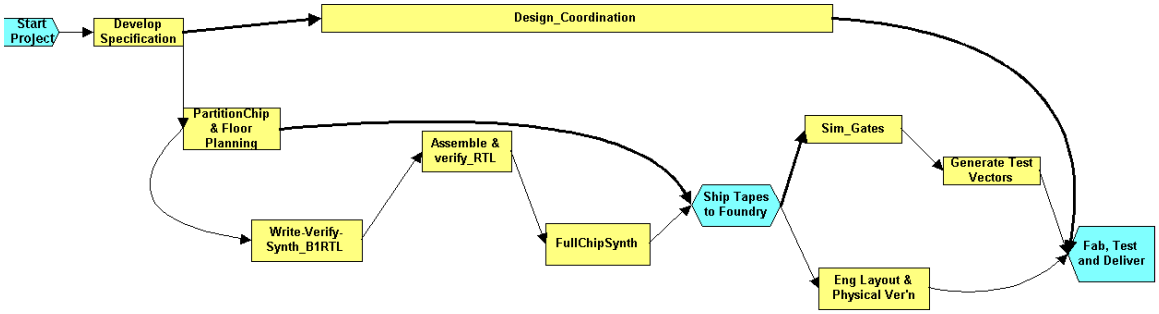

Figure 3: Traditional CPM diagram in Vite

The sample demo is illustrated as shown in Figures 4, 5 and 6. Figure 4 shows the Ghant chart for the Vite project activtities shown in Figure 3. Vite to PSL(KIF) translator generates a PSL file, and the scheduling information is produced by P3 as shown in Figure 5. Using the P3 to PSL(KIF) translator, the project information in P3 (as shown in Figure 5) is parsed to generate a PSL file. Finally, using PSL(KIF) to MS Project translator, we can reconstruct the project in Microsoft Project and produce Gantt chart shown in Figure 6.

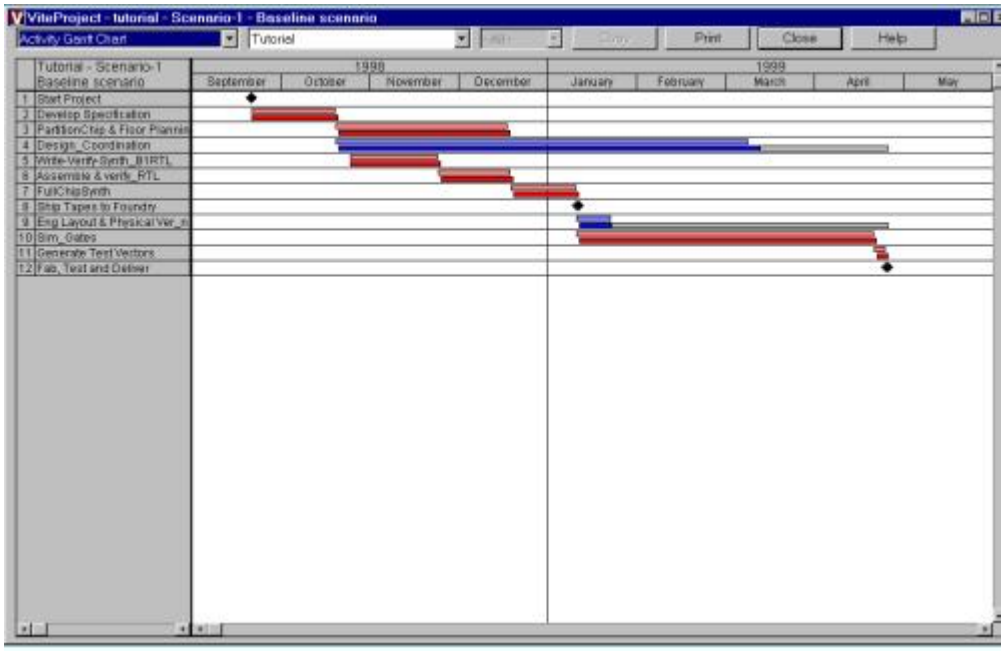

Figure 4: Gantt Chart in Vite

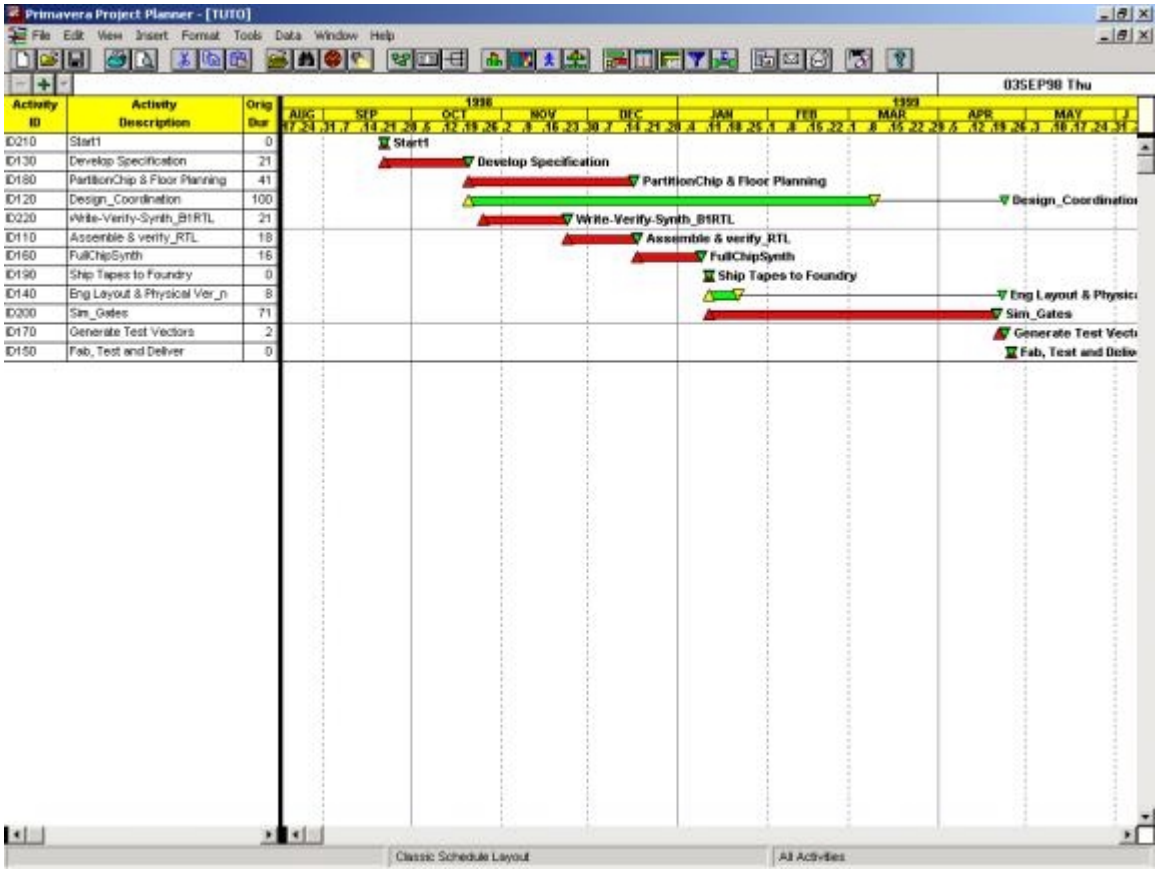

Figure 5: Gantt Chart in P3

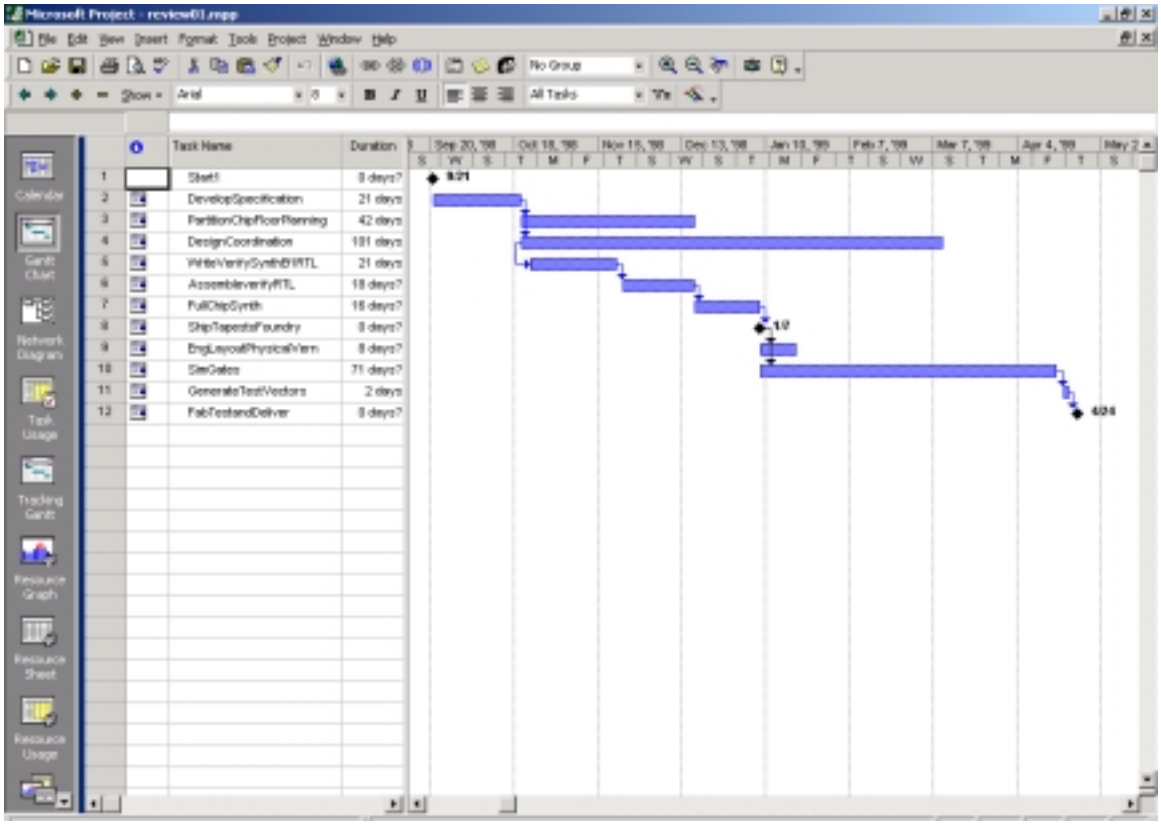

Figure 6: Gantt Chart in Microsoft Project 2000

Comparison Between PSL and aecXML

We have conducted a comparison study between PSL and aecXML. Below is a summary of comparison results. Detailed discussion on the comparison is currently under preparation.

The goal of PSL is to create a process interchange language that is common to all manufacturing applications, generic enough to be decoupled from any given application, and robust enough to be able to represent the necessary process information for any given application. This representation would facilitate communication among the various applications because they would have a common understanding of the concepts to be shared.

AecXML is an XML-based language used to represent information in the Architecture, Engineering and Construction (A/E/C) industry. The information includes resources such as projects, documents, materials, parts, organizations, professionals or activities such as proposals, design, estimating, scheduling and construction. It is intended to be used as an XML namespace and to facilitate information exchange of A/E/C data on the Internet. The main idea with aecXML is not only to establish some standard ways of structuring building data but also to enable automated processing of the data as much as possible.

Both PSL and aecXML can be used to exchange project scheduling information. Generally speaking, PSL expressions are more general and fundamental, while aecXML expressions are more direct, explicit and convenient for construction project scheduling. PSL has better expression power than aecXML.

PSL(KIF) is based on first order logic and situation calculus, and all concepts in PSL are formally defined using KIF . Based on axioms in PSL and reasoning rules, it is possible to carry out reasoning based on existing knowledge. AecXML (and XML in general) is a markup language and is not capable of specifying the semantics of the terms that it introduces. On the other hand, it is also more difficult to build a generic PSL parser than an aecXML parser.

PSL and aecXML have different emphases. PSL is designed primarily for manufacturing industry, while the target audience for aecXML is A/E/C industry. PSL and aecXML each can be more efficient in certain areas. For example, PSL can be used to exchange knowledge among different programs and perform reasoning with existing knowledge. XML files are semi-structured data; thus, aecXML is more suited for storing and sharing project information similar to a database system. Also, aecXML can be used to wrap the query results from database and transfer the results onto Internet applications.

Currently aexXML(XML) gets more attention than PSL(KIF). However, it is not to say that aecXML will dominate in A/E/C industry. Most likely, several ontology will coexist in A/E/C industry, such as STEP, PSL, IFC and aecXML. There will be some translators among these ontologies, which can map from one ontology to another.

PSL integration with SimQL

SimQL consists of the basic environment and SimQL Schema/Query Language to reuse the simulation results. The SimQL environment contains the following five models:

- SimQL Server
- Wrappers
- Simulator Interface
- SimQL Agent/Query Programs
- Simulators

OKBC(Open Knowledge Base Connectivity) provides a set of operations for a generic interface to underlying KRSs(knowledge representation systems). OKBC is complementary to language specifications developed to support knowledge sharing. PSL(KIF) provides a declarative language for describing knowledge. As a pure specification language, PSL(KIF) does not include commands for knowledge base query or manipulation. However, OKBC focuses on operations that are efficiently supported by most KRSs (such as query operation and manipulation operation).

As the picture (Figure 7) illustrated below, SimQL sever could receive query command from the user, and invoke relative simulators (such as PS, MS Project and Vite). The simulators send the simulation results back to the SimQL sever (in PSL format). Then the SimQL sever could store the simulation results into the knowledge sharing system through OKBC server. Once the simulation results are stored in knowledge sharing system, it could be shared by different programs, and users could even do some reasoning on the knowledge.

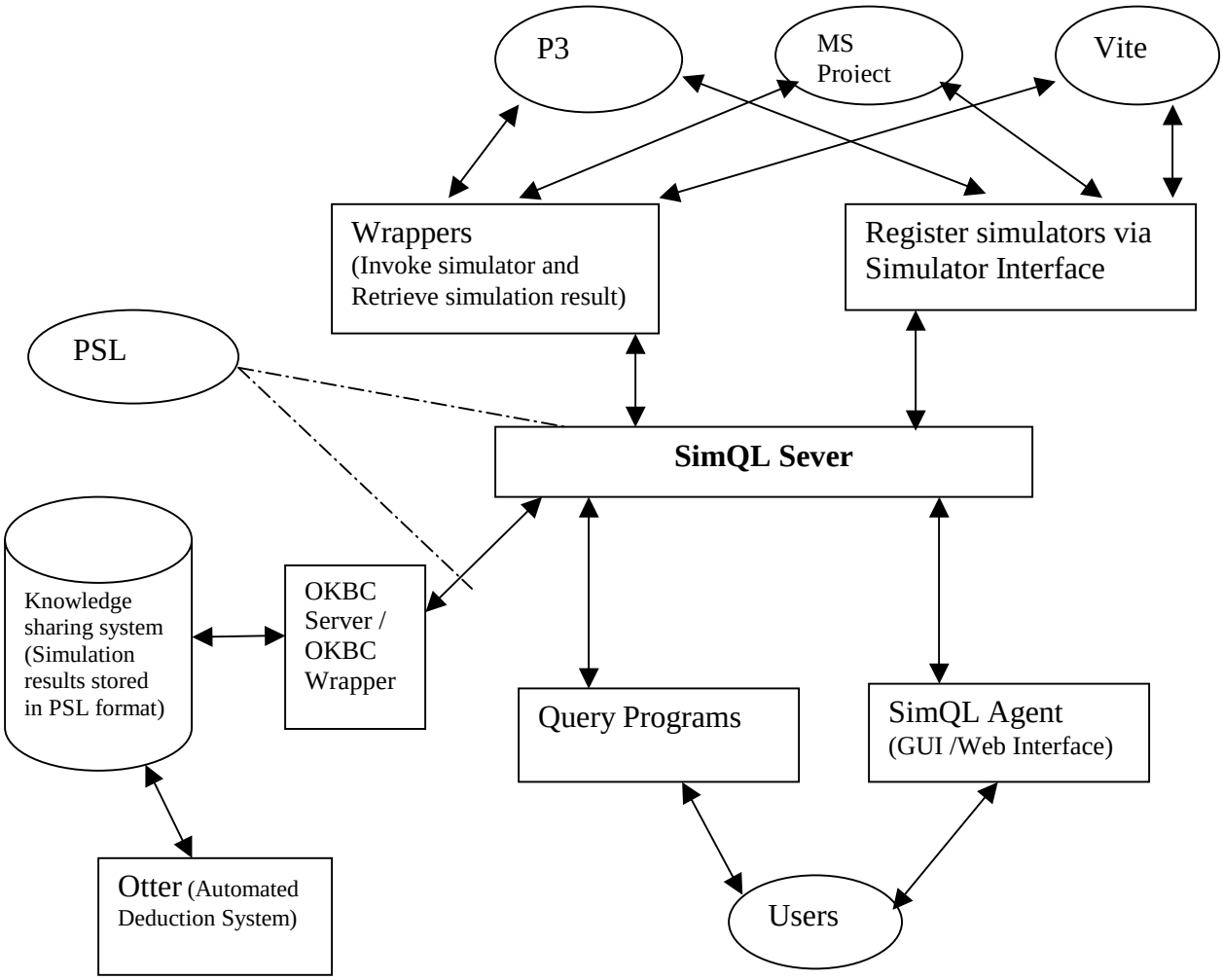

Figure 7: PSL integration with SimQL