U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology

NISTIR 5318

National PDES Testbed
Report Series

APDE Demonstration System Architecture

Stephen Nowland Clark
Allison Barnard Feeney

NATIONAL PDES TESTBED

NIST

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology

NISTIR 5318

National PDES Testbed

APDE Demonstration System Architecture

Stephen Nowland Clark
Allison Barnard Feeney

U.S. DEPARTMENT OF COMMERCE
Ronald Brown,
Secretary of Commerce

Technology Administration
Mary L. Good,
Under Secretary for Technology

National Institute of Standards and Technology
Arati Prabhakar, Director

January 14, 1993

Table of Contents

Disclaimer

No approval or endorsement of any commercial product by the National Institute of Standards and Technology is intended or implied.

Acknowledgement

The National PDES Testbed project is funded by the Office of the Defense CALS (Continuous Acquisition and Life-Cycle Support) Executive.

APDE Demonstration System Architecture

Stephen Nowland Clark

Allison Barnard Feeney

1 Introduction

The emerging international Standard for the Exchange of Product Model Data (STEP[1]) comprises six distinct types of specifications[2]. "Implementation Specifications" provide descriptions of mechanisms for the actual exchange of STEP data (e.g., a clear text exchange file format [7]). "Description Methods" provide techniques for specifying STEP (e.g., the EXPRESS information modeling language [6])."Integrated Resources" provide information models describing generic constructs which are useful in a wide variety of product descriptions (e.g., geometry [8]). "Conformance Testing" Parts and "Abstract Test Suites" deal with defining the rules for establishing conformance to STEP. "Application Protocols" are the Parts of STEP which combine components of Integrated Resources, select implementation mechanisms, and use the Description Methods to specify what product data is to be exchanged and the meaning of that data in a particular industrial context (e.g., associative draughting [9]). In essence, Application Protocols (APs) are the Parts of STEP which are implementable. Thus it can be expected that CAx[3] vendors will provide mechanisms in their products which will facilitate data exchange according to particular APs. A thorough introduction to STEP and its constituent specifications can be found in "Overview and Fundamental Principles" [5].

The National PDES Testbed program at the National Institute of Standards and Technology is focused on the development and implementation of STEP. Principal funding for the National PDES Testbed program is provided by the Office of the Defense CALS (Continuous Acquisition and Life-cycle Support) Executive. There are several projects within the National PDES Testbed; among these is the effort to establish an Application Protocol Development Environment (APDE), a software environment supporting the information processing needs of STEP AP developers.

---

1. STEP is being standardized under the auspices of the International Organization for Standardization (ISO) Technical Committee 184 (TC184) Subcommittee 4 (SC4). The term PDES (Product Data Exchange using STEP) refers to the United States' effort contributing to this standardization process.
2. STEP will be released as a collection of specifications; each individual specification is known as a "Part" of STEP.
3. The term "CAx" refers to any type of engineering, manufacturing, or operations software application system, e.g., Computer-Aided Design (CAD), Computer-Aided Process Planning (CAPP), etc.

The APDE will consist of a set of software tools which will aid in the efficient construction of Application Protocols for STEP. While much work has been done in the way of general purpose tools which are applicable to various portions of the AP development process (such as office automation, technical documentation, information modeling, and project management), the NIST APDE project is unique in that it aims to establish an integrated, dedicated environment specifically geared to support the AP development process as a whole.

Prior reports have documented high-level requirements and functional specifications for the full APDE [1][4]. The current report documents the architecture of a prototype system demonstrating a portion of the APDE. When possible, the relationships between this prototype architecture and the architecture ultimately envisioned for the APDE are noted. The demonstration system will illustrate a portion of the functionality envisioned for the APDE, primarily focusing on those aspects of the AP development process for which automation is seen as having the largest near-term payoff.

1.1 The AP Development Process

The purpose of this section is to give an overview of the AP development process. For complete details, the reader is urged to consult the AP guidelines document [16]. It is important to note that no commercial software tools are currently available which are specifically intended to facilitate AP development. AP developers currently use an ad hoc collection of document processing software, information modeling software, and virtually anything else available which may make their task easier.

1.1.1 AP Project Definition

An AP project is initiated by documenting, in English prose, an industry need for the AP, thereby establishing the requirement for a particular AP in STEP. A high-level statement of scope is agreed upon (and updated as the AP becomes better defined). In order to further document the industry need, the AP developers produce an Application Activity Model (AAM) which specifies the processes that use and produce product data in the context of a specific application. The AAM is documented using IDEF0 [19] methodology. Once a comprehensive AAM is developed, each element of the AAM is examined and a determination is made whether the element is in or out of scope, based on the intended use of the AP. The scope statement, the completed AAM, and a Candidate AP Summary sheet are submitted for approval as an ISO AP project.

Once the AP project is approved, the scope, requirements, and AAM are evaluated by experts in the application area who were not involved in the initial modeling effort. These experts should reflect the breadth and depth of the application scope. The AAM is modified to ensure that it accurately represents industry processes and practices, accurately reflects the desired scope, and is correct and complete. The results of the industry review are documented separately in the AP Validation Report.

1.1.2 AP Information Requirements

After the AP scope has been defined and evaluated, the information requirements of the AP are defined through the development of an Application Reference Model (ARM). The ARM is documented using one of three graphical data modeling languages (EXPRESS-G[1][6], IDEF1X [20], or NIAM [15]). The model diagrams are a required informative annex of the AP, and the information requirements are normatively described in text. Each element in the ARM diagram is defined as an Application Object in the AP. Each relationship between elements in the ARM diagram is documented as an Application Assertion. The concepts in the ARM are organized into Units of Functionality (UoF). A UoF is a grouping of constructs which reflect one or more distinct concepts within the ARM, possibly corresponding to an application process. The UoFs are potentially useful for evaluating areas of commonality between APs.

The ARM must be evaluated by industry experts as was the AAM. The objective of ARM validation is to provide a high degree of confidence that the model supports industry practices correctly and robustly. It is impractical to conduct a comprehensive review of the ARM due to its complexity. The validation is done with the use of representative test pieces and usage scenarios. The model may be validated by any of several methods. One method is to build a prototype database that replicates the structure of the ARM [14]; another is to perform paper populations of the structure and requirements. The method used to perform ARM validation is documented in the AP Validation Report, along with the results of the validation.

1.1.3 AP Interpretation

The Application Interpreted Model (AIM) is developed by mapping elements from the STEP Integrated Resource (IR) Parts to meet the information requirements described in the ARM. This process, known as Interpretation, may result in further constraints on IR constructs and the creation of new constructs in the AIM when an appropriate IR construct is not available for a particular ARM requirement.

The interpretation process requires cooperation between those who have developed the ARM and others who have extensive knowledge of the Integrated Resources. Interpretation is typically carried out in a workshop-style meeting. Detailed notes from this meeting are compiled into an Interpretation Report that becomes a part of the AP Validation Report. Another output of the Interpretation workshop is a Mapping Table which shows the correspondences between the elements of the ARM and those from the Integrated Resources.

The AIM is documented by an EXPRESS information model known as the Short Listing. The Short Listing consists of references to Integrated Resource elements and definitions of any new elements and constraints added during interpretation, e.g., specializations of, and rules that further constrain, Integrated Resource elements.

---

1. EXPRESS-G is a graphical subset of EXPRESS. See Annex D of [6] for a description.

An AP also contains an expanded form of the Short Listing, known as the Annotated Listing. The Annotated Listing contains the complete documentation of the AIM. This includes textual descriptions of all of the Resource elements used and other constructs added during interpretation, as well as the EXPRESS definitions of these elements. During the development of the Short and Annotated Listings, the AP developers must also develop a complete graphical representation of the AIM as well, using EXPRESS-G.

Finally, as with the ARM, validation of the AIM must be performed and documented in the AP Validation Report. AP usage information formulated for the validation process may also be provided as an informative portion of the AP document itself.

1.1.4 Complete AP

Once the AP developers have completed the documentation and validation of the AIM, the remainder of the AP development work involves defining implementation and conformance requirements, and completing the AP documentation. The conformance and information requirements ensure that there are metrics available against which vendor implementations of the AP can be tested for conformance. The information requirements and assertions defined in the ARM and all characteristics defined in the AIM are the starting points for the development of such conformance requirements. Test Groups are defined from the structure of the ARM and Test Purposes are defined for all constructs of the AIM and documented in an Abstract Test Suite. Review and evaluation of the AP's Conformance Requirements and Abstract Test Suite is performed by application experts and AP methods experts. The results of this evaluation are included in the AP Validation Report.

AP developers are also responsible for compiling all of the requisite components of the AP specification into a document according to established style guidelines [18]. At several stages of AP development the document is submitted to various committees and representatives of voting members (countries) in ISO for review and comment. Throughout the development and balloting process, AP developers must maintain logs of issues raised against the AP and the team's responses to these issues.

1.2 The APDE of the Future

The Application Protocol Development Environment (APDE) will be a tightly integrated software environment supporting the AP development process outlined above. Wherever possible, the APDE will make use of commercial off-the-shelf (COTS) software, but some tools will need to be NIST-developed. The goals of the APDE are to drastically reduce the time required to build an AP, and to improve the quality of the resulting AP. The cornerstone of the APDE will be the AP Information Base (APIB), a repository containing the information content of APs under development as well as various STEP documents relevant to the development of APs.

One aspect of the APDE is its support for the various processes which develop the actual information content of the AP, e.g. the various information models and activity models which must appear in the AP. Thus, the APDE will include such software as information model editors, requirements tracking tools, and configuration management

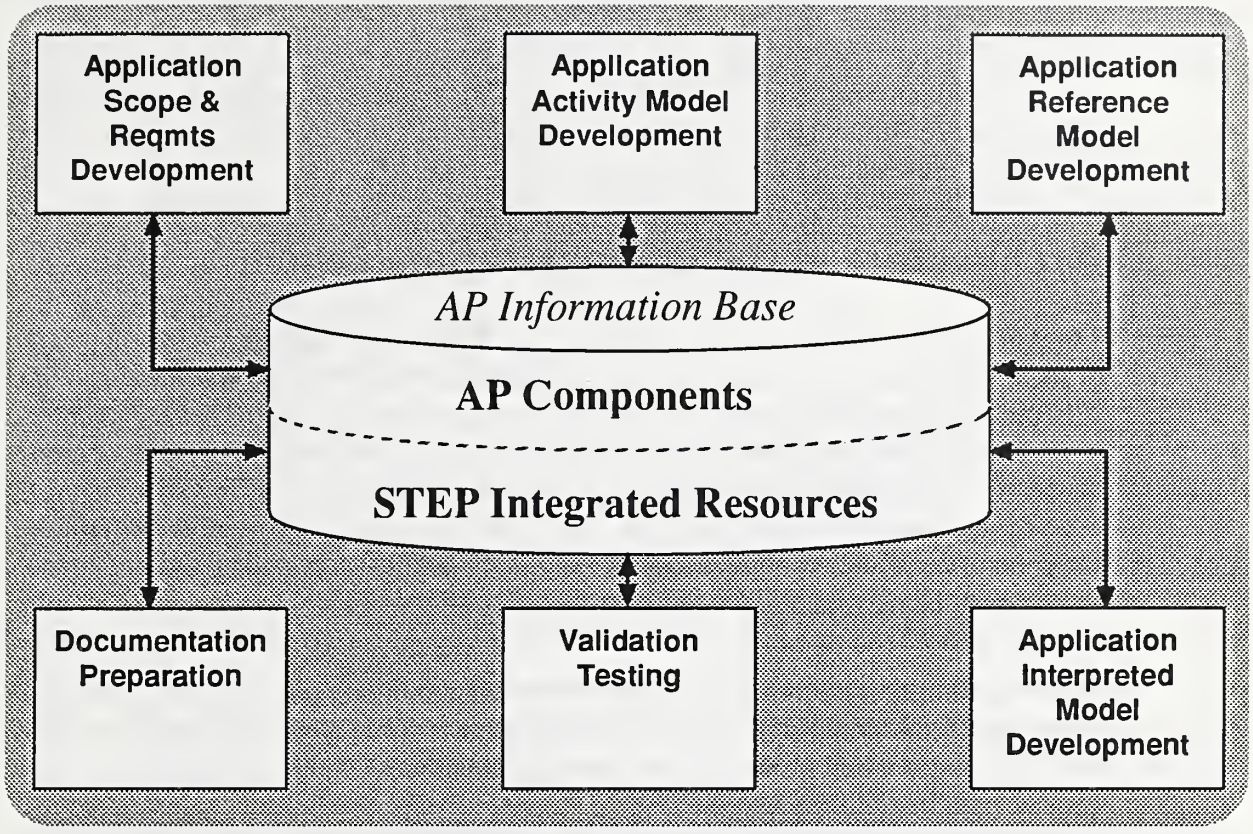

Figure 1: Future APDE Architecture

tools. Another equally crucial aspect is the support provided by the APDE for the documentatation of these requirements in the form required for standardization. Thus, the APDE must also include tools supporting such things as desktop publishing and enforcement of ISO and STEP style and layout rules.

A high-level architecture of this future APDE is shown in Figure 1. Various modules will address Scope and Requirements development, AAM development, ARM development, AIM development, ARM and AIM Validation testing, and Documentation preparation.

The tools making up the APDE will communicate via the AP Information Base (APIB). This database will store STEP documents (Integrated Resources, APs, and eventually even methods documents) in a highly structured manner. For example, individual subclauses corresponding to particular definitions found within the information models in an AP will be explicitly represented in the APIB. To this end, all of the documents stored within the APIB will be tagged using Standard Generalized Markup Language (SGML)[10]. SGML is an international standard which allows the formal specification of a document's structure. When a document is carefully tagged using SGML, structural tags can be associated with content. One of the benefits of this approach is that it allows software systems to recognize and manipulate information based on its content.

Once these STEP documents have been tagged using SGML, they will be stored in a powerful database. The various components of the APDE will then be able to make sophisticated queries against this APIB, greatly easing the task of developing the content of an AP. By explicitly representing the structure of the documents it contains, the APIB will also provide support for a publishing application which can assemble an AP from its constituent parts according to the editorial guidelines specified by ISO and STEP, and thus produce a correct AP document.

2 Architecture of the Demonstration System

The APDE demonstration system will focus primarily on the AIM development portion of the AP development process. This focus has been chosen for several reasons. Interpretation is widely viewed as the largest single hurdle in the AP development process. It is often described as a black art, and requires intimate knowledge of all of the Integrated Resources as well as the ARM under Interpretation. Software support, even for the more mundane aspects of Interpretation, is critically needed. In addition, the generation of the Annotated Listing from the Short Listing lends itself to automation. Finally, an AIM validation tool, *Data Probe* [13], has already been developed at NIST.

The demonstration system will include facilities for building and browsing the Short and Annotated Listings of the AIM. An IR browser will be provided to aid in the ARM-to-AIM interpretation process and to aid understanding of the AIM once it is constructed. In addition, an EXPRESS-G browser will be included, representing the ARM development module. *Data Probe* will be integrated into the system to support the AIM validation process.

2.1 The APIB in the Demonstration System

The Application Protocol Information Base (APIB) is envisioned as a central STEP document repository on which the rest of the APDE will be built. Although the APIB will eventually make use of a powerful database management system (DBMS), the demands of this early demonstration system are not so great as to require a DBMS.

For the purposes of this demonstration system the APIB will consist of a set of files within the UNIX file system. There are several types of files to be considered. These include ARM EXPRESS-G files, Integrated Resources, AIM Short Listings, and AIM Annotated Listings. The APIB will provide the ability to store and retrieve each type of file. In the future, the APIB will support operations which extract various pieces of these files, and so will have to understand the internal structure of each. In the demonstration system, however, most of this functionality is already embedded within the systems which will make up the APDE. Until this functionality can be migrated from these end-user components, there is little need for any more flexibility in the APIB itself.

2.1.1 ARMs (EXPRESS-G)

The demonstration system will support only one ARM representation, namely EXPRESS-G. In the future, it is intended that the APDE will support IDEF1X and NIAM as well, as STEP allows any of these three representations to be used for ARM development.

The demonstration system will make use of a commercially available EXPRESS-G browser. The format of ARM EXPRESS-G files in the demonstration APIB will be determined by the selection of this browser. EXPRESS-G browsers typically can import and export EXPRESS files, but use a proprietary file format to represent EXPRESS-G models complete with layout information.

2.1.2 Integrated Resources (SGML)

During the past year, the APDE project has developed an SGML Document Type Definition (DTD) representing STEP Integrated Resource Parts [3]. Integrated Resources in the demonstration system will be SGML documents tagged using this DTD. This will allow the system to demonstrate the value of a tailored DTD for a specific class of STEP documents.

2.1.3 AIM Short and Annotated Listings (HTML)

In the future, all of the documents in the APIB will be represented using SGML, to facilitate the representation of their internal structure. To date, however, there is no DTD available for STEP Application Protocols; there is only one for IRs. In order to demonstrate some features of SGML which currently are not exploited by the DTD used for the IRs, the AIM Short and Annotated Listing files in the demonstration APIB will be tagged using HTML (HyperText Markup Language). HTML is a particular form of SGML which has been developed for the World-Wide Web project [2]. HTML provides a primitive DTD which can be applied to any type of document. However, the tags defined by HTML are geared towards establishing hypertext links between documents rather than capturing the structure of those documents. Thus, although it provides a useful demonstration of the hypertext-like capabilities of SGML markup, it is not a complete solution to the problem of tagging STEP documents to capture their structure.

2.2 Demonstration System Components

The APDE demonstration system will consist of a collection of loosely connected tools, integrated primarily by the fact that they will share data files through the UNIX file system. Although these various components will be more tightly integrated in the APDE of the future, the nature of this demonstration system and of the tools from which it will be built requires this loose coupling.

The high-level architecture of the demonstration APDE is shown in Figure 2. Scope and Requirements development and AAM development will not be addressed in this initial prototype; rather, they are assumed to have taken place outside of the context of the

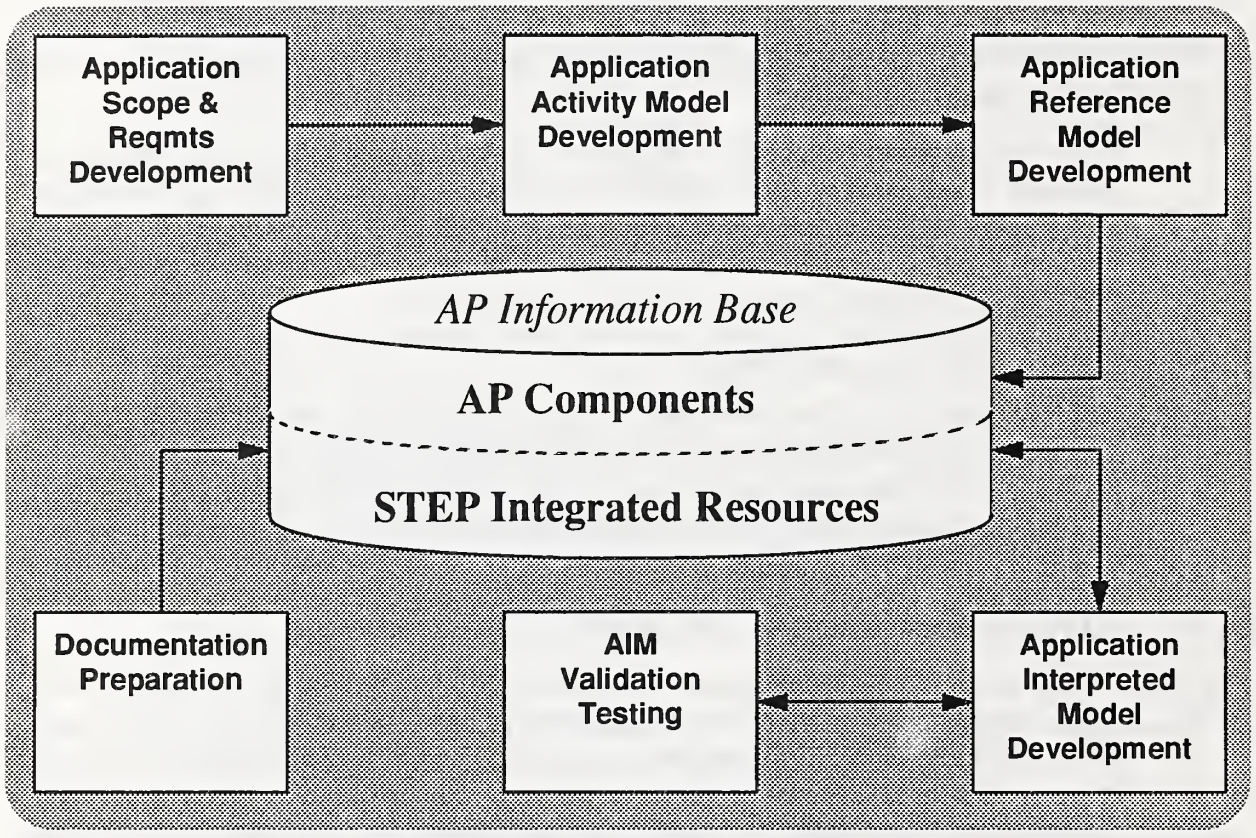

Figure 2: Preliminary APDE Architecture

APDE. Primitive facilities will be provided for ARM development and Documentation Preparation, and more comprehensive tools will address AIM development and Validation.

The major components of the demonstration system are shown in Figure 2, and are discussed in more detail below.

2.2.1 Integrated Resource Browser

Although it is not visible in the high-level APDE architecture diagram, the Integrated Resource Browser is an important component of the APDE. It will support the browsing (and, eventually, querying) of STEP Integrated Resource Parts, an important aspect of the AIM development process and a very useful aid in the understanding of an existing AIM.

Each IR document will be tagged in SGML using the DTD which has been developed for the APDE project. This tagging allows the structure of an IR document to be explicitly represented, down to the level of the subclauses which contain individual EXPRESS type and entity definitions. These SGML tags will provide the IR Browser with sufficient information to perform such operations as browsing a particular clause of interest (e.g., Definitions) or finding the EXPRESS definition of a particular entity or type of interest (e.g., Circle).

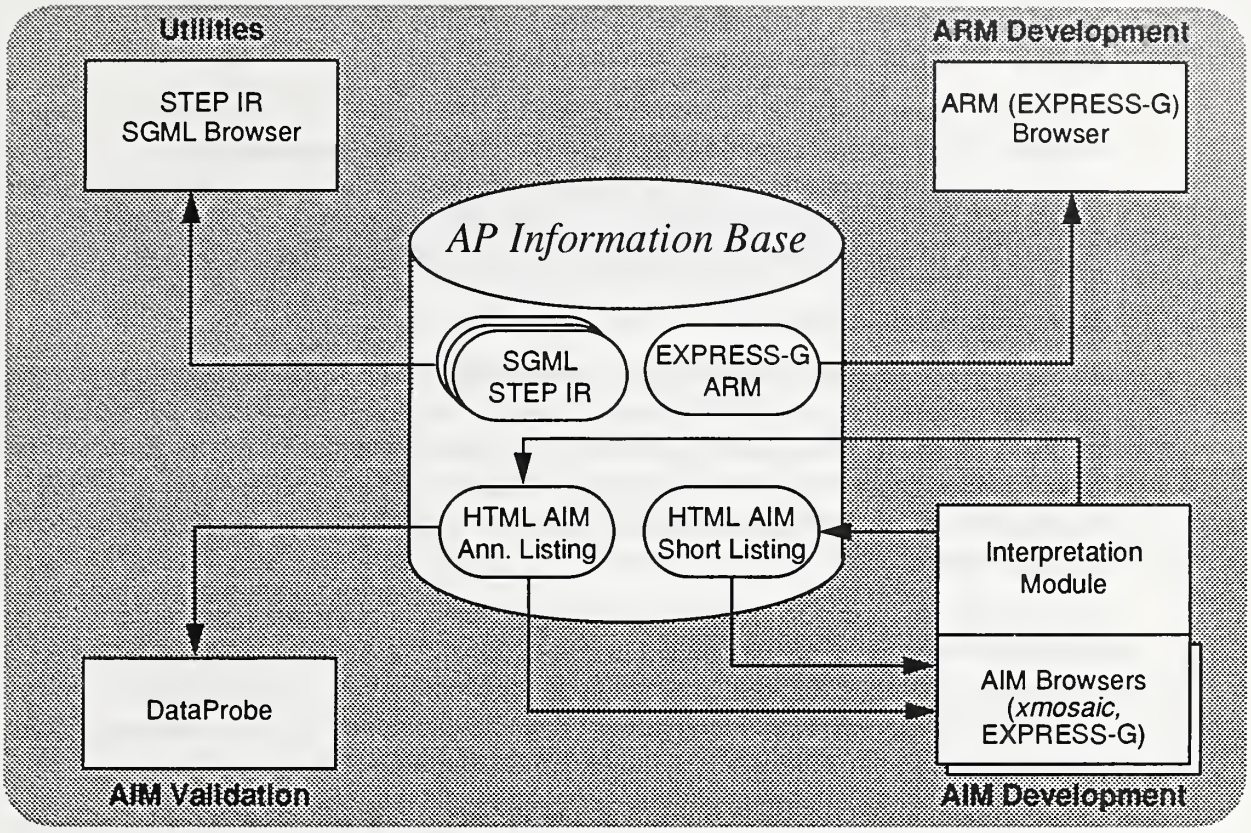

Figure 3: APDE Demonstration System Components

The browser itself will be based on a commercial SGML browsing tool, which has yet to be selected. This tool must have the ability to browse documents tagged with an arbitrary DTD, because the DTD used for these documents has been developed specifically for the APDE project and is not standardized[1]. No editing facilities will be provided by this tool; modifying Integrated Resource Parts is beyond the scope of the APDE.

In addition to its browsing capabilities, the browser must be able to start up at, or remotely caused to move to, an arbitrary position in the document browsed, because of the way that this browser will be invoked from the other components. This may be done using Unix Remote Procedure Calls (RPC) or a tool such as *expect* [11], a NIST-developed tool for managing control and communications between processes.

2.2.2 Application Reference Model Browser

This tool will support the display of ARMs developed using EXPRESS-G. No editing facilities will be provided: although future plans for the APDE include support for ARM development in EXPRESS-G as well as NIAM and IDEF1X, this demonstration

1. There is significant interest in SGML within the STEP community, and our DTD will provide useful input should the community choose to adopt SGML for future development of STEP.

system focuses on the AIM development process. AIM development requires the existence of an ARM, and so the ARM Browser will be provided so that the user is able to view the ARM as it is mapped into the AIM.

The ARM Browser will be a commercial EXPRESS-G tool, which has not yet been selected. This tool will provide basic browsing functionality, including panning/scrolling, zooming, and searching for particular constructs by name. Although EXPRESS-G tools typically use their own proprietary formats to store EXPRESS-G models, the tool should be able to directly import EXPRESS models as well.

2.2.3 Interpretation Module

The Interpretation Module will support the construction of AIM Short Listings and the subsequent automatic generation of AIM Annotated Listings from them. Because there are no commercial tools available to directly support this process, this module will be built by integrating and extending existing tools. These tools include *shtolo*, a NIST-developed tool for producing Annotated Listings, and *Tk* [16], an X Window System programming toolkit.

This module will support primitive querying of the information models in the Integrated Resource Parts contained in the APIB, and will allow particular definitions from these models to be interfaced in the Short Listing. As the Short Listing is being built, HTML tags will be inserted into the document to maintain cross-references to the sources of these definitions in the Integrated Resources.

In addition to direct references to Integrated Resource definitions, a Short Listing may contain other EXPRESS constructs which define specializations of these definitions or further constrain their use within the AP's context. To support the definition of these constructs, the Interpretation Module will include a primitive EXPRESS editor which will allow EXPRESS definitions and associated annotations to be entered into the Short Listing.

Once the Short Listing has been completed, the Interpretation Module will be able to generate the complete EXPRESS form of the corresponding Annotated Listing. This will be done using the NIST EXPRESS Toolkit [12]. The first step of this process is to verify that the Short Listing is a valid EXPRESS model, and to allow the user to correct any errors found using the EXPRESS editor.

Although STEP requires that the Annotated Listing include far more documentation than simply the complete EXPRESS model, the initial implementation will not provide this additional documentation. The complete annotation for each Resource definition will be present in the Integrated Resource document in which the definition originally appears. The Annotated Listing, like the Short Listing, will contain HTML tags cross-referencing the Integrated Resource source for each of these constructs. In this way, the full annotation of constructs originating in the Integrated Resources will be available via the AIM Browser (see section 2.2.4) and the IR Browser.

In addition to the Short and Annotated Listings, the AIM is required to be documented in EXPRESS-G. The Interpretation Module will make use of the ARM EXPRESS-G browser's ability to import EXPRESS and layout an EXPRESS-G diagram to provide the ability to automatically produce this AIM Diagram.

2.2.4 Application Interpreted Model Browser

The AIM Browser will allow both the Short Listing and the Annotated Listing of an AP to be browsed. These two AP components will be tagged using HTML, an SGML variant developed for the World-Wide Web project. This will allow the AIM Browser to be built using *xmosaic* [2], an X Window System-based HTML browser.

A feature of HTML which is particularly useful is that it is able to represent hypertext links between documents. These links can then be traversed by *xmosaic*. Thus, with the Short and Annotated Listings tagged using HTML, it will be possible to select a particular entity definition in the AIM, and have the Integrated Resource Browser automatically start up viewing the definition of that entity in the Integrated Resource Part from which it is drawn. This cross-document linking ability will be an extremely important feature of the APDE.

In addition to this *xmosaic*-based browser for the Short and Annotated Listings, the EXPRESS-G browser used in the ARM development module will be available at this stage in the process as well, allowing the AIM Diagram to be browsed, but not edited. In the future, it will be necessary to allow changes to the Diagram, e.g. to rearrange page layouts and such, but this need will not be addressed in the demonstration system.

2.2.5 Validation Module

After developing the AIM, the AP development team must validate this model, to ensure that it meets the industry requirements to be addressed by the AP. One way of performing this validation is to build sample populations of the model to represent real-world industrial scenarios. This type of validation is supported by *Data Probe*, an X Window System-based application developed at the National PDES Testbed. *Data Probe* will be integrated into the demonstration APDE, so that sample populations of the Annotated Listing of the AIM can be built.

3 Conclusion

The current state-of-the-art in software support for Application Protocol development lags far behind what might reasonably be expected in such a high-tech, computer intensive standardization effort as STEP. The APDE project attempts to address this situation by building a coherent and integrated set of tools to support the many aspects of AP development.

The APDE demonstration system described in this report is intended to illustrate how a more complete APDE might function in the future. Important aspects of this demonstration system would likely be duplicated in a complete system: SGML tagging to capture the structure of STEP documents; a central repository of STEP documents

and APs-in-work; a variety of modules supporting various stages in the AP development process, all communicating via this repository; and shared software resources such as the IR Browser supporting multiple stages in the process.

While a significant portion of the AP development process is given little or no attention in this demonstration system, the tools described which focus on the ARM to AIM mapping portion of the process are expected to be of significant value to AP developers. These tools will provide a solid foundation for future work in expanding the APDE to address the entire AP development process.

A References

[1] Barnard Feeney, A., Clark, S. N., and Fowler, J. E., Requirements for an Application Protocol Development Environment, NISTIR 5197, National Institute of Standards and Technology, Gaithersburg, MD, May, 1993.

[2] Berners-Lee, T.J., Cailliau, R., Groff, J-F and Pollermann, B., World-Wide Web: The Information Universe, in *Electronic Networking: Research, Applications and Policy*, Vol. 2 No 1, pp. 52-58, Meckler Publishing, Westport, CT, Spring 1992.

[3] Bodarky, S., and Paisley, S.W., An SGML DTD for the STEP Integrated Resource Parts, NISTIR 5224, National Institute of Standards and Technology, July, 1993.

[4] Clark, S.N., Barnard Feeney, A., and Fowler, J.E., Specifications for an Application Protocol Development Environment, NISTIR 5248, National Institure of Standards and Technology, Gaithersburg, MD, August, 1993.

[5] ISO/DIS 10303-1, Industrial automation systems and integration – Product data representation and exchange – Part 1: Overview and fundamental principles, International Organization for Standardization, March 26, 1993.

[6] ISO/DIS 10303-11, Industrial automation systems and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual, International Organization for Standardization, August 31, 1992.

[7] ISO/DIS 10303-21, Industrial automation systems and integration – Product data representation and exchange – Part 21: Implementation methods: Clear text encoding of the exchange structure, International Organization for Standardization, May 21, 1993.

[8] ISO/DIS 10303-42, Industrial automation systems and integration – Product data representation and exchange – Part 42: Integrated generic resources: Geometric and topological representation, International Organization for Standardization, April 29, 1993.

[9] ISO/CDC 10303-202, Industrial automation systems and integration – Product data representation and exchange – Part 202: Application protocol: Associative draughting, International Organization for Standardization, February 2, 1993.

[10] ISO/TR 9573-11, Information Processing – SGML Support Facilities – Techniques for using SGML – Part 11: Application at ISO Central Secretariat for International Standards and Technical Reports. First Edition, International Organization for Standardization, 1992.

[11] Libes, D., Expect: Curing Those Uncontrollable Fits of Interaction, in *Proceedings of the Summer 1990 USENIX Conference*, pp. 183-192, Anaheim, CA, June 11-15, 1990.

[12] Libes, D., The NIST EXPRESS Toolkit – Introduction and Overview, NISTIR 5242, National Institute of Standards and Technology, Gaithersburg, MD, August, 1993.

[13] Morris, K.C., Data Probe: A Tool for EXPRESS – based Data, *Proceedings of the Seventh Annual ASME Database Symposium - Engineering Data Management: Key to Success in a Global Market*, American Society of Mechanical Engineers, New York, August 1993.

[14] Morris, K.C., Mitchell, M.J., and Sauder, D., Validating STEP Application Models at the National PDES Testbed, NISTIR 4735, National Institute of Standards and Technology, Gaithersburg, MD, December 1991.

[15] Nijssen, G.M., and Halpin, T.A., Conceptual Schema and Relational Database Design: A Fact Oriented Approach, Prentice Hall, 1989.

[16] Osterhout, J. K., An Introduction to Tcl and Tk, draft, Addison-Wesley, 1993.

[17] Palmer, M., and Gilbert, M., Guidelines for the development and approval of STEP application protocols, Version 1.1, ISO TC184/SC4/PMAG Technical Report, August, 1993.

[18] Shaw, N., ed., Supplementary directives for the drafting and presentation of ISO 10303, ISO TC184/SC4, Editing Committee Draft N20, February 11, 1993.

[19] Specifications for Integration Definition for Function Modeling (IDEF0), Proposed Federal Information Processing Standard, September 9, 1992.

[20] Specifications for Integration Definition for Information Modeling (IDEF1x), Proposed Federal Information Processing Standard, September 9, 1992.