STEP SOFTWARE ARCHITECTURES FOR THE INTEGRATION OF KNOWLEDGE BASED SYSTEM

Patrice Poyet, Guillaume Besse, Eric Brisson, Philippe Debras, Alain Zarli* and Jean Luc Monceyron **

ATLAS (EP 7280) is a collaborative research project to improve the competitiveness of European Large Scale Engineering companies within the world market. The project focuses on the development of software Architectures and Tools based on non proprietary information exchange standards such as STEP and EDIFACT to improve business integration and co-ordination. The understanding of software integration does not only cover classical engineering applications but also addresses knowledge based systems (i.e. rule based and case based) by means of STEP based AI environments including a LISP binding of the SDAI specification (STEP Part 22). The work undertaken during the course of the project opens the road to the future integration of KB systems within STEP architectures.

Key words: STEP, Software Integration, Knowledge Based Systems, Large Scale Engineering.

ATLAS INTEGRATION STRATEGIES

ATLAS strongly rely on the concept of semantic integration (Brisson et al., 1993), (Böhms et al., 1994b). This means that the integration between different software applications is performed by means of a suite of reference models. The modelling methodologies used by the ATLAS project to create these reference models are rather traditional in the STEP community.

Let us remind the reader that STEP (ISO 10303) is an International Standard for the computer-interpretable representation and exchange of product data. The objective is to provide a neutral mechanism capable of describing product data throughout the life cycle of a product, independent from any particular system. The nature of this description makes it suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases and archiving.

* CSTB - Centre Scientifique et Technique du Bâtiment, BP 209, 06904 Sophia Antipolis Cedex, France, e-mail: poyet@cstb.fr
** ENTPE - Ecole Nationale des Travaux Publics de l'Etat, Rue Maurice Audin, 69518 Vaulxen Velin Cedex, France. e-mail: monceyro@medoc.entpe.fr

In that context, NIAM (Nijssen's Information Analysis Method) is mainly used for data modelling and IDEF0 for the functional and the business analysis (i.e. building construction and process plants design and installation) (Böhms et al., 1993a-b). The formal information modelling language EXPRESS supports the formal representation of the models which are organized according to a suite ranging from the more generic (LSE Project type Model) (Tolman, 1993) to the more specific known as View type Models « VtMs » (Tolman et al., 1994a-b) (e.g. Architectural VtM (Böhms et al., 1994a), HVAC VtM, Acoustic VtM (Monceyron et al., 1994a), etc.). Provisions have been made to support a complete mapping of NIAM models into EXPRESS constructs and vice-versa.

These models can be directly implemented by a new generation of software applications and communication is then straightforwardly achieved or the existing models used by current applications (referred to later on as Application type Models - AtM) can be translated (syntaxic level) and converted (semantic level) whenever necessary to support the exchange with the software systems compliant the open systems approach.

Up to now we have encountered two different sets of models. The first set comprises the data models allocated to applications called Application Type Models (AtM) in compliance with the ATLAS terminology. The AtM therefore describes the structure of the internal Model within an application. It is upon this model, together with much more information, that the application runs its functionality.

Within the ATLAS terminology there are two more levels in the modelling structure: the Large Scale engineering Model (LSE) is the most generic model, which will cover very abstract concepts. The first specialisation of this model is the Building Project Type Model (BPtM). This model is then again specialised in the different View type Models (VtM).

In figure 1 the dependencies between these models are shown. On the highest level one has the Large Scale Engineering (LSE) model. On the next level the Building Project type Model (BPtM) and the Process Plant Project Type Model (PPPtM) are found. These two models are specialisations of the generic LSE model. One level further down one finds a large number of View type Models (VtMs). On the same level of the VtMs one finds the Application type Models (AtMs). The difference between the two, from a pragmatic point of view, being that the VtM should be neutral towards specific applications and the AtMs be not known to anybody apart from the application programmers.

View type Models (VtM) are linked to particular life cycle stages of the project and this approach leads to the definition of a bi-dimensional matrix where the actors represent one axis and the life cycle stages another one. For each one of the cells of the matrix there exist a specific VtM at a particular project life cycle stage. Of course, other axis representing different concepts could be envisaged and would extend the number of cells to be considered (e.g. systems leading to System type Models or StM).

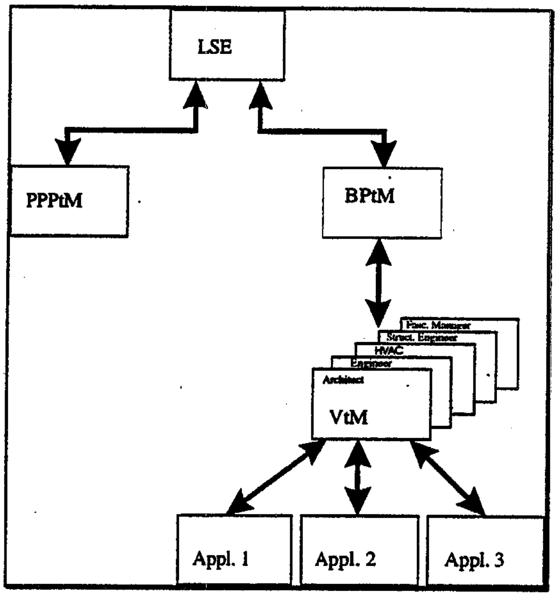

Figure 1: The dependencies within the ATLAS models.

IMPLEMENTATION TOOLS

Developing models is one thing implementing these models is quite another. In fact, the usage of a given set of models is already one way of achieving some level of inter-operability between a variety of software applications. The communication process between the business actors can then rely for example on the exchange of neutral files (e.g. SPF files) (ISO 10303 - Part 21) and communications barriers are overcome.

But, rapidly this first objective appears to fall short of the real services expected. In fact, a common data storage, taking charge of the concurrent access, distribution and sharing of data across networks and business partners is the level of requirement that the end-users place on open systems integration technologies. Even though STEP does not fully address in the current initial release all these requirements, it nevertheless brings practical routes to implementations strategies.

During the course of the ATLAS project a number of language bindings of the SDAI specification (Part 22) have been implemented. Some have been developed as extended facilities of data management systems such as the OMS of SNI, or created for specific purposes such as the integration of knowledge base systems (LISP binding).

Applications making use of these bindings have the capability to manipulate data through the interface as if it is represented in the underlying data storage technology exactly as it is described in the defining EXPRESS schema or schemas.

A complete software architecture supporting concurrent engineering still need to be worked out by STEP. Further work is planned to address knowledge modelling extending the realm of product / project modelling into a wider perspective and to support complex business processes requiring distribution, sharing and concurrent access to common repositories. In some respect these problems have been considered out of the scope of the ATLAS project, even though the LSE industry is definitely a good example of a potential sector where these techniques would have a great impact and a clear benefit on the companies' working practice. In that respect ATLAS belong to the « club » of concurrent engineering projects supported by the EC.

INTEGRATION OF KNOWLEDGE BASED SYSTEMS

Rule Based Systems

Knowledge Based Systems are getting acceptance in the industry and major applications, especially in the process plant sector, are being operated by end users. But from the very beginning of the development of Artificial Intelligence (AI) techniques one of the major problem has been the disconnection of AI based applications from the remaining software components routinely run by end-users. This comes from the very nature of AI systems which require different programming environments and software engineering techniques.

This situation is extremely detrimental for the industry as the need for intelligent systems is recognized and as these KB applications cannot operate most of the time as standalone environments. This means that using expert systems has often been considered in the industry as an expensive process which finds its justifications only if there is no other way out ! This obviously prevents the widespread use of knowledge based applications and deprives the industry from the benefits it might expect from advanced software solutions.

ATLAS argues that an alternate route for the integration of knowledge based systems is made possible by the work undertaken in the STEP arena. Instead of expecting that the integration will arise from the uniform usage of particular programming language(s) considered as the widely accepted industrial standard (e.g. C++) the assumption defended in this paper is that STEP offers a suitable frame and provides standardized modelling languages, methodologies and implementations techniques which can also serve the needs of KB systems, still leaving to the implementor the freedom of the final target languages (e.g. LISP as an example of SDAI binding).

For the ATLAS project, the problem of the integration of knowledge based systems has been addressed as a matter of proper software infrastructure for the development of knowledge based systems. An SDAI late LISP binding has been implemented and a complete implementation platform based on such a binding has been prototyped. This includes complementary facilities such as an EXPRESS compiler, SPF (i.e. Part 22) readers / writers, a user interface to handle the SDAI environment, etc. The set of ATLAS reference models can then be loaded into such platform and manipulated by means of the binding.

Of course, appropriate extensions have been developed to provide support for the reasoning functions, for the representation of control structures and as a long term objective for the modelling and exchange of knowledge. Reasoning relies on a rule based language. This language has its own grammar which defines what are the properties and the expressiveness of this language. The rules are then parsed by an appropriate compiler and the code generated for the execution of these rules makes use of the late LISP binding so that the rules can operate atop of the SDAI storage(s).

Finally a control structure offers means to organise the reasoning process. The rules are arranged into rule bases, in turn associated to tasks representing elementary set of actions to be performed to undertake the reasoning process. The complete problem solving strategy is based on a sort of divide and conquer strategy which breaks down recursively into subtasks the tasks which cannot be solved at a given level of the reasoning process. Many additional parameters have actions on the reasoning process, such as controlling the unification technique used, activating the hypothetical reasoning facilities (i.e. shall concurrent states be created ?), memorising the reasoning process (i.e. supporting backtrack facilities), etc.

Examples of knowledge based systems have been developed by means of this software architecture and the potential of these KB techniques with respect to model conversion problems has been evaluated. For example, the conversion of building projects made of entity instances conforming to the Speedikon AtM into entity instances of the ATLAS neutral Architectural VtM has been experimented and demonstrated. This attempt was very successful and demonstrates the interesting potential of rule based techniques not only for the traditional development of intelligent applications but also for the reasoned conversion of semantic models.

Further work is being undertaken in the area of building acoustics by ENTPE in order to enlarge the suite of ATLAS reference models with an acoustical view type model and to make use of this model to implement, by means of the software infrastructure developed during the course of this task, an expert system for checking the compliance of any residential building project against the regulation.

Case Based Systems

LSE projects are by their nature normally non-repetitive. They are, nevertheless, assembled from collections of components and units used in previous projects. Some are standard (e.g. propietary products and process systems) whilst others are common generic solutions (e.g. reinforced concrete beams, steel pipes).

In most advanced projects there is a degree of innovation, however, which solves a particular local need but which should be stored for a potential use on fututre projects as a previously developped technical solution.

For the life-cycle integration of product/process information we need a mechanism for data-abstraction and -generalization. Information created during the production and operation/maintenance stages of a product life cycle is not directly usefull for re-use in future projects, especially if this information has to be used in an early design stage.

Information about realized products and processes has to be abstracted and generalized, partially by application of statistics. The resulting information can be archived, updated by using new actual information, and applied in new projects. The information provided by this mechanism is company or discipline related. It contains general, re-usable information about aspects of a product/process.

The technical approach explored to make use of past designs is Case Based Reasoning (CBR). CBR techniques have been developed by the AI community and are increasingly used by practical applications running in operational environments.

The route followed has been to use the semantic integration principle (the suite of ATLAS product / project models) and the SDAI architecture to develop a CBR application generator, i.e. XP-CB. Specific applications, addressing the LSE industry needs can then be further developed. An example will be considered to illustrate the potential of these techniques.

XP-CB is a generator of CBR applications and provides various archiving and retrieval facilities. XP-CB is an open shell and is based on an object oriented implementation. The entity definitions of the conceptual model of XP-CB can be further specialized. The functionalities have been implemented as messages and can also be specialised. This implementation enables the user to satisfy specific needs which are out of scope of a generic shell.

A case is seen as a network of STEP instances. As matter of fact, the basic functionalities are to archive and to retrieve STEP instances. A network of instances means an instance with its set of attributes, some of them referring to other instances, the complete structure representing a network. All the functions of the API of XP-CB use a late LISP binding so that the archived cases can be retrieved from an SDAI storage. A Case Data Base consists of a collection of STEP instances of having the same entity definition.

Two major methods are proposed to archive and retrieve a case. In the first one, cases are stored by means of an indexation tree. This tree is built from criteria defined by the developer and based on the model of case. This method ensures a very efficient retrieval but hardly support incomplete data.

The other method supported is similarity measure based on the definition of a distance between the cases. To get a similar case from the data base to a given one, this method computes the distance between the case submitted and all the cases of the database and retains the case which gives the lowest distance. This method is less efficient than indexation trees but fully supports incomplete data.

A combination of these two methods is often the best solution: an indexation tree is used to retrieve a first set of similar cases. The similarity measure is then applied to retrieve the most similar cases.

The two main concepts used by XP-CB are classification and partition :

- a classification is a set of ordered criteria defined for an entity definition;

- a partition is an indexation tree derived from a classification : it is a memory structure to archive cases.

XP-CB consists of a conceptual workbench and of a library of functions for the runtime usage of the partitions (atop of a C binding of the SDAI specification):

- the conceptual workbench is used to define one or several classifications and to generate the corresponding partition in a SPF format. Before generating and saving the partition, you can test and validate the classification / derived partition by means of archiving and retrieving trials within the conceptual environment.

- the C functions of the runtime library have to be integrated within the CBR application. These functions support the archive and retrieval operations by processing the partition generated before hand.

USING THE SOFTWARE ARCHITECTURE

An Acoustics KB System for Building Design

An acoustics knowledge based system was developed, making use of an acoustics VtM designed according to the ATLAS methodologies (Monceyron et al., 1995), and implemented by means of the software architecture described in the previous section. The main goal of this development is the elaboration of a knowledge based system for the automatic assessment of the acoustics quality of building projects.

ENTPE had already developed in the past various intelligent systems in this specific expertise area, the essential weakness of which was the inability to communicate easily with other tools (such as CAxx systems) or to import or export building project data according to a neutral format. In this context, a STEP based approach appeared to be a promising route to provide for these missing communication capabilities, therefore ensuring a better integration of knowledge based systems both at the modelling and at the implementation level (including the SDAI specification).

The first step has consisted in the definition of an Acoustics View type Model (VtM), following the ATLAS modelling methodologies, making use - throughout a mapping relationship - of the Building Project type Model. This acoustics VtM describes the universe of discourse of the acoustics engineers (both activities and data flow) throughout the life cycle of building projects. An activity model has been produced using the IDEF-0 methodology recommanded by ATLAS, and then an information model, required for the representation the acoustics viewpoint has been developed.

About the acoustics VtM, it is worth noticing that it is not a VtM exactly in the same sense as the other VtMs, like HVAC or Process Engineer for instance, which reflect a specific view of an actor in the building branch. In fact, this Acoustics VtM cuts across other VtMs. The acoustics VtM adds to or ignores features of pre-existing entities always defined in the scope of other VtMs. Consequently, the Acoustics VtM appears as the result of a « mechanism » extracting information from the other models.

Once defined, the Acoustics VtM provides a frame describing the structure and the semantics of the information involved in building acoustics, acting as the core of a knowledge based system. A prototype of such a system (referred to in the text as XP-ACOU) was then developed by means of XP-SDAI™ (model management facilities), XP-KB™ (mastering the reasoning process) and essentially XP-RULE™ for all the aspects of conversion between VtMs and rules based evaluation(s).

In its current development stage, XP-ACOU is a knowledge based expert checking the compliance of apartment buildings against the French acoustic regulation (recently published early 1995). All the verifications are made according to the Qualitel method or to the new French acoustics regulation (NRA) : it compares the performances of a given configuration with the regulation's requirements, generates a new design when the evaluation is unsuccessful, assesses its (generation - evaluation - correction cycle paradigm), and may produce a final brief to summarise the progress and results of the expert system. XP-ACOU is entirely written using the XP-RULE™ language.

To give an idea of the usefulness of such knowledge based environments for the practitionner exchanges between a widely available CAD system (Autocad) XP-ACOU have been implemented. This is why the whole process is in fact divided in several steps making use of a variety of systems (some expert) :

- a layer (written in Autolisp), to translate the entities defined in Autocad in a STEP physical file format according to the Architectural VtM ;

- a completion expert (written in XP-RULE™ and using hypothetical reasoning of XP-KB™ under some hypothesis), in order to automatically complete the semantic information contained by the Architectural model. This operation is required because the objects are not defined at the same level of semantics by Autocad and by the Architectural model ;

- a conversion tool (written in XP-RULE™) to support the mapping between objects of the Architectural VtM and objects of the Acoustics VtM ;

- the aforementioned knowledge based, i.e. XP-ACOU (generate - evaluate - correct paradigm), supplemented with another conversion tool (also written in XP-RULE™) for the inverse mapping from the Acoustics VtM to the Architectural VtM ;

- a translator (written in XP-RULE™) from the Architectural model to the DXF format to allow at any step of the process the visualisation by means of Autocad of the improvements or modifications performed on the model by the acoustics expert. To do this, the DXF entities have been objectified, deriving an EXPRESS schema for DXF.

All these experts are extensively described in the Annex 3 of this deliverable. They show how a number of concepts and technologies may be intermixed to offer a STEP compliant knowledge based system for the acoustics domain.

CONCLUSIONS

Semantic models should serve as a neutral and reference core supporting the data exchange mechanisms. Applications can make use of these models in various ways depending on how far they "integrate" these models at the level of their internal application models. If the applications make direct use of the reference model (e.g. which can be a STEP AP), and provided that they benefit from a STEP pre and post file processor to support file exchange capabilities, they will be integrated along the data dimension.

To open the door to a full integration these applications would also have to make use of a SDAI interface to get access to the data stored in the common repository. This requires an in depth modification of the software concerned and this migration cannot be achieved overnight. Nevertheless, to ensure an overall control over the usage made of the applications and of the data generated in order to move toward some sort of concurrent engineering processes, a central data storage supporting SDAI appears as a requirement.

However, the success of this semantical integration scenario relies on a major assumption: the wide and rapid use of a limited number of standard reference models (i.e. STEP Aps). On more practical ground, one might consider that the main factor on which depends the success of the integration along the data axis is the capability to develop, rapidly and at a limited cost, intelligent and versatile converters able to map the different application semantic models.

In that respect, rule base systems also offer the potential not only to serve as application domain software supporting specific business functions, but can also play a major role with respect to the development of versatile, easy to use - easy to modify - rule based converters and model mappers. This one of the areas where the ATLAS project has used these KB techniques for its own internal needs, i.e. mapping semantical models by reasoning.

REFERENCES

(Böhms et al., 1993a) M. Böhms, J. Neuteboom, G. Storer, J. Stephens, J.L. Fidalgo, 1993. ATLAS Deliverable D101 : Functional Analysis, Business models for the existing situation of the Building Industry (TW) and Process Plant Industry (Centunion), CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1100, issue Jan. 1993, restricted.

(Böhms et al., 1993b) M. Böhms, J. Neuteboom, A. Eijs, W. Bakkeren, R. Greening, G. Storer, J. Stephens, J. Kattenhorn, Angel Gayoso, P. Poyet, 1993. ATLAS Deliverable D102 : Data Analysis, CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1200, issue Nov. 1993, restricted.

(Böhms et al., 1993c) M. Böhms, R. van Mieghem, E. Brisson, 1993. ATLAS Deliverable D103 : Modelling Methodology for Open Systems Integration, CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1300, issue Dec. 1993, restricted.

(Böhms et al., 1993d) M. Böhms, R. van Mieghem, P. Poyet, E. Brisson, P. Debras, 1993. ATLAS Deliverable D104a : Software tools for modelling supporting the Methodology for Open Systems Integration, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1400, issue June 1993, restricted.

(Böhms et al., 1993e) M. Böhms, A. Eijs, W. de Bruijn, 1993. ATLAS Deliverable D105a : Guidelines for the EXPRESS to Implementation Formalism Generation, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1500, issue June 1993, restricted.

(Böhms et al., 1994a) M. Böhms, F. Tolman, R. Greening, J. Stephens, 1993. ATLAS Deliverable D106-IIIa : View type Model for the Architect in the Global Design Stage, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1500, issue Dec. 1993, restricted.

(Brisson et al., 1993) E. Brisson, P. Poyet, 1993. Implementation of the COMBINE Integrated Data Model. Final Report to the CEC, March 15th, 1993.

(Debras et al., 1991) P. Debras, P. Poyet, E. Brisson, 1991. Expert Systems and Documentary Databases. Micro-computers in Civil Engineering, An International Journal - Elsevier Science Publishers, Vol 6, n° 4, 1991, p. 281-289.

(Debras et al., 1993) P. Debras, P. Poyet, 1993. ATLAS Deliverable D402a : Information Management and Document Handling System, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP4, Task 4200, issue Oct 1993, restricted.

(Greening et al., 1994a) R. Greening, P. Poyet, R. Singh, M. Böhms, G. Storer, 1994. ATLAS Deliverable D201 : LSE Project Modelling Kernel, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP2, Task 2100, issue Jan. 1994, restricted.

(Greening et al., 1994c) R. Greening, U. Frey, P. Poyet, 1994. ATLAS Deliverable D501a : ATLAS Prelinimary Project Results, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP5, Task 5400, issue Feb 1994, restricted.

(ISO 10303 - Part 11) ISO/DIS 10303 - 11 : Industrial Automation Systems and Integration - Product Data Representation and Exchange - Part 11 : Description Methods : The EXPRESS language reference manual, August 1992

(ISO 10303 - Part 21) ISO/DIS 10303 - 21 : Industrial Automation Systems and Integration - Product Data Representation and Exchange - Part 21 : Clear text encoding of the exchange structure, August 1992

(ISO 10303 - Part 22) ISO/DIS 10303 - 22 : Industrial Automation Systems and Integration - Product Data Representation and Exchange - Part 22 : Implementation Methods : Standard Data Access Interface, ISO TC 184/SC4/WG7 N302 May 1993

(ISO N51,1994a) "EXPRESS-M Reference Manual", Product Data Representation and Exchange, ISO TC184/SC4/WG5 N51, January 1994

(ISO N51,1994b) "EXPRESS-M Reference Manual", Product Data Representation and Exchange, ISO TC184/SC4/WG5 N51, May 1994

(Monceyron et al., 1994a) J.L. Monceyron, J. Beaumont, P. Poyet: Acoustical View type Model, ENTPE / LASH, France, 39 p.

(Monceyron et al., 1994b) J. L. Monceyron, P. Poyet, J. Beaumont. A view dependent model for acoustics in building design with communication purpose. Proc. of the 1st ECPPM Conference, Oct 1994, Dresden, Germany.

(Palmer, 1993) M. Palmer, M. Gilbert, 1993. Guidelines for the development and approval of STEP application protocols, version 1.1.

(Poyet et al. 1994a) P. Poyet, E. Grivart, E. Brisson, G. Besse, M. Irvine, R. Greening, M. Böhms: ATLAS Deliverable D301a : Implementation of Knowledge Base Extensions, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP3, Task 3100.

(Poyet et al. 1994b) P. Poyet, E. Brisson, E. Grivart, J. van Horssen, B. Behage, M. Mooij, R. Greening, R. Singh: ATLAS Deliverable D302c : Data Abstraction Generalisation and View Conversion Mechanisms, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP3, Task 3200.

(Singh et al., 1993) R. Singh, Soon Leah Foo, M. Böhms, G. Storer, A. Gayoso, 1993. ATLAS Deliverable D404a : Data and Knowledge Manager, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP4, Task 4400, issue Dec 1993, restricted.

(Soon Leah Foo, 1993) Soon Leah Foo, 1993. ATLAS Deliverable D401a : ATLAS LSE Application Platform - First Version, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP4, Task 4100, issue June 1993, restricted.

(Tolman, 1993) F. Tolman, 1993. ATLAS Deliverable D106-Ia : LSE Project Reference Model, Requirements, Assumptions and Basic Constructs, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1500, issue June 1993, restricted.

(Tolman et al., 1994a) F. Tolman, W. Bakkeren, M. Böhms, 1994. ATLAS Deliverable D106-Ib : ATLAS LSE Project type Model, Generic Part of the ATLAS LSE Project Reference Model, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1500, issue Jan 1994, restricted.

(Tolman et al., 1994b) F. Tolman, M. Böhms, W. Bakkeren, 1994. ATLAS Deliverable D106-IIIa: ATLAS Building Project type Model, ESPRIT III - CIME IV.2.3, Project # 7280 - ATLAS, WP1, Task 1500, issue Jan 1994, restricted.