CORE
Metadata, citation and similar papers at core.ac.uk
Provided by NASA Technical Reports Server

GRANT
IN-89-CR
329080
P7

# Hubble Space Telescope Design Engineering Knowledgebase (HSTDEK)

## Final Report

by

Dr. James D. Johannes
Clark Everetts

of

The University of Alabama in Huntsville
Computer Science Department
Research Institute M34-C
Huntsville, AL 35899

for

National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Huntsville, AL 35812

under
Contract Number NAG8-117
Account Number 5-31976

(NASA-CR-187838) HUBBLE SPACE TELESCOPE DESIGN ENGINEERING KNOWLEDGEBASE (HSTDEK) Final Technical Report, 12 Sep. 1988 - 11 Sep. 1989 (Alabama Univ.) 7 p CSCL 03A

N91-17845

Unclas
G3/89 0329080

## 1. Introduction and Research to Date

The overall objective of the Hubble Space Telescope Design Engineering Knowledgebase (HSTDEK) project is to develop techniques to incorporate knowledge engineering into the traditional engineering activities associated with large NASA system development efforts. The short term research to support this goal focuses on the development of methodologies for building large, multipurpose knowledgebases, and tools to enhance the knowledge capture effort during the design phases of large NASA space systems.

The first phase of this HSTDEK research effort concentrated on current documentation production and management. Various types of knowledge were identified and a knowledge taxonomy was formed. The NASA system development process used in the design and construction of the Hubble Space Telescope was analyzed in order to gain understanding of a typical development effort and to identify baseline release dates and frequency of revision of specific documents and document types. This was done to find points in the design process where the process itself could benefit from knowledge capture taking place. The documents were classified into categories describing the types of knowledge that could be expected to be obtained from them, and were assessed in terms of their utility for knowledge capture and knowledge engineering. This work is described fully in Research Progress Report 1 [3].

The second phase of research had as its focus the development of ways to map design knowledge into a knowledge base. Knowledge representation methods were evaluated in light of their ability to represent the various types of knowledge identified during phase one. An object-oriented knowledge representation method was selected as the most appropriate for the types of knowledge that would exist in the knowledge base, and an example knowledgebase was developed based on this paradigm. This work is documented in Research Progress Report 2 [4].

The contractual research covered in this report pays specific attention to the development of tools to 1) assist knowledge engineers in acquiring knowledge and 2) to assist other technical, engineering and management personnel to automatically perform knowledge capture as part of their everyday work without adding any extra work to what they already do. Requirements for data products, the knowledge base, and methods for mapping knowledge in the documents onto the knowledge representations will be discussed as will some of the difficulties of capturing in the knowledge base the structure of the design process itself along with a model of the system designed. The capture of knowledge describing the interactions of different components will also be briefly discussed.

## 2. Recommendations for Data Products

For documentation and other data products to be 'amenable' to automated knowledge acquisition, there are some factors which should be considered. These include: the media upon which the data product is stored, the form or structure of the information in the data product, a means of indexing or referring to the data product (i.e., identifying clearly and completely the sources of the information in the data product and how it was derived), the frequency of change and the methods of change control, internal and external consistency, correlating 'related' information, highlighting references to information that contradicts information in a data product, documentation formatting standards and automated document formatters (templates for reports, for example).

The media upon which a data product is available directly affects the way in which knowledge contained in the data product can be extracted. Information produced (or only generally available) in paper must undergo some transfer to electronic storage, either through direct keyboard entry into the knowledge base, or by optically scanning the document and importing the desired information into the knowledge base. The problem with the first approach is obvious: re-keying even a small fraction of the volume of documents produced by NASA and its contractors is simply not feasible. The second approach is not really much better, because it is difficult to extract and save in the knowledge base only the information that is useful, and to create the necessary links to related information. It is clear, then, that some form of capturing the information while it is still in electronic form is imperative.

Highly structured information is more easily captured and interpreted than loosely structured information. Examples could be tabular data or sets of information represented similar to a record structure. Structured information is more easily captured because it is simpler to choose what is useful information and what is not useful. Structured data is also more easily accessed physically.

## 3. Consistency and Change Control

There are several types of consistency that should be maintained in the knowledge base, in terms of the information itself as well as its storage and use. Internal consistency refers to the information contained within a data product, while external consistency refers to resolving contradictions between information contained in different data products represented in the knowledge base. For examples of automating internal consistency checking, consider the addition of a test report to the knowledge base.

For what reasons besides the obvious must consistency be maintained? What are the major approaches to enforcing consistency; i.e., how can consistency be maintained, and how can consistency enforcement be automated? What types of systems or information are inherently more difficult to manage in terms of consistency? Conversely, what inherent qualities of systems make them more or less manageable? What are some potential problems/'things' that negatively influence consistency? To limit the repetition of knowledge; this is necessary to conserve space and to help make consistency less of a problem. An audit trail for the knowledge is necessary; this is difficult when the knowledge sources are diverse in type.

Form or structure consistency is also important. The knowledge base should represent the same type of knowledge in the same way each time it is added to the knowledge base; this is difficult with multiple knowledge engineers and requires strictly enforced guidelines on knowledge representation.

To review, some potential types of knowledge sources include: English text documents, diagrams, schematics, figures, photographs, computer-generated numerical or text data, test reports, audiovisual recordings (of voice, image, or computer data), and optical mass storage technology. Develop specific guidelines for how different types of knowledge should be represented.

The Acceptance Test Data Package that is submitted at the Configuration Inspection Review(s) contains a great deal of experiential knowledge including test results and nonconformance reports. sampling of HST data products that appear to be valuable sources for design knowledge capture. NASA projects are developed over a long period of time using a wide variety of knowledge types from many disciplines. Agfa Compugraphic's CAPS Author/Editor (Implementation of SGML portion of CALS). Consistency in usage of terms; create a knowledge dictionary and project taxonomy.

**4. Arguments for an Object-Oriented Knowledge Representation**

Research Progress Report 2 [4] contained some arguments for using an object-oriented approach to knowledge representation. Discussed here are some more additions to the rationale. First, such an approach can model a system using isa and part-subpart links, or other relations, more easily. It can include methods to retrieve information from other sources, to invoke other programs for simulation or graphical display, or other traditional programs. This would allow diagrams to be associated with objects which in turn would allow for more effective presentation of information. More efficient programming results through the use of inheritance. A rule base could access objects, and rules could more easily be partitioned. In an object-oriented environment, procedures can dynamically create objects as needed. Correlating related information should be easier in an object-oriented environment as should be emphasizing contradictory data found in the knowledge base.

The most interesting and perhaps difficult aspect of a large multipurpse knowledge base is the idea that it would provide for the smooth integration of vast amounts of information. It may be necessary for the KB to be partitioned, so that memory and

Advanced automation techniques should be used in knowledge acquisition when possible. Techniques for integrating automation into the knowledge acquisition process should be studied; different types of knowledge acquisition may require the use of different knowledge acquisition tools. To illustrate this

A knowledge engineer must be consistent in the way that he maps knowledge into knowledge base objects and in the way he or the expert system shell manages the interaction between those objects. It is more important, even vital, that standards for knowledge representation and procedures for updating the knowledge base be followed when there are a large number of knowledge engineers involved in knowledge acquisition efforts for a large NASA system. Ways should be found to automate the process so the knowledge engineer is burdened less with having to manually ensure that information is being entered properly into the knowledge base.

Need to specify how knowledge base will be structured early on.

## 5. Recommendations for the Knowledge Base

The knowledge base should be kept smaller rather than larger. For knowledge base development environments that require the entire knowledge base to be in memory this is particularly important. A smaller knowledge base is more easily managed (and modified) and will be faster than a larger. A single multipurpose knowledge base will require a tendency toward structured information. Starting the knowledge capture effort too early can result in inefficient use of time because much of the design is not well defined. Because may significant design changes are likely to occur early in the design process, it is possible that a knowledge base would be built for a component that is never actually constructed. Of course, such a knowledge base used for simulation may help determine that a particular component is inappropriate for the task for which it was intended.Many current expert system shells operate by placing the entire knowledge base in main memory. This simply will not do for a large, multipurpose knowledge base of the size envisoned by How well do expert system shells make use of secondary storage? KEE and ART make the entire knowledge base memory-resident; this simply will not do for a knowledge base of the size envisions.

Some assumptions about the knowledge base have to be made, or some facts have to be known before it is designed and implemented. For example, what types of applications will use the knowledge base? It is necessary that this be known for purposes of verification & validation, for limiting the amount of information that must be represented and to determine the granularity of the knowledge to be included, and to determine if the "right" system is being developed and to determine if it is being built correctly ::is this not just another view of V&V??:: Some potential uses for the knowledge base have been discussed in [3] and [4], and include people looking up information or identifying problems; expert systems to access the knowledge base ::obvious::; to incorporate knowledge into or form a foundation for other systems, such as simulations or tutorials.

## 6. Role of the knowledge engineer

In order for knowledge engineers to be accepted as an important part of the traditional design process, engineers and designers much see a benefit from having knowledge engineers involved. Knowledge engineers must demonstrate that they can assist engineers in traditional engineering activities (skepticism has been noted in interviews; not necessarily resistance, but engineers have have an honest perception that there is no place for AI in their work - not because there is anything "wrong" with it, but rather "there simply isn't anything it can do for us in our work"). One area where immediate benefits would be most beneficial to traditional engineers and would get them accustomed to AI techniques would be documentation. Make documentation easier to produce, make it more complete, make it easier to access and maintain documentation, and engineers will begin to see that advanced automation techniques do indeed have a place in their work. The emphasis on practical tools is very important.

Knowledge engineers should be involved in the earliest stages of a system development effort in order to gain familiarity with the project and the people involved, and to identify potential knowledge sources. Because of the degree of change that occurs in the initial design during the period just after the RPF and phase A work, actual knowledge capture may not be useful until later. After the design has become somewhat more firm, during phase B, for example, the knowledge engineers' work would really begin. It would be beneficial to have the knowledge base constructed soon after this because, as the design of the system changed, the knowledge base would provide an automatic, complete, and more usable record of the state of the design. Starting knowledge capture too late leads to the difficulties NASA knowledge engineers are experiencing now; knowledge is lost over time. This implies the need for KEs to be a part of the design process in very early stages. This will help them gain ::understanding of the problem::, develop approaches for solving it, and::areas of high payoff/interest::.

## 7. Suggestions for Tools

Tools have been alluded to in the previouis progress reports. Some of these included intelligent interfaces to math models, tutorial systems with an explanation facility

::what are the general motivations for providing an explanation capability?
::Is an explanation capability necessary for various tools used by NASA personnel and contractors?::
::What kinds of tools would benefit from the incorporation of an explanation component?::

The requirements for some documents may have to be changed in order to facilitate the use of its information in a knowledge based system. At the very least, documents should be submitted in electronic form. Current paper documentation should be translated to electronic form if it cannot be obtained directly from contractors in that form. Therefore, practical uses of scanning technology should be made to effect the translation of paper documentation into electronic form.

## 8. Summary

Apart from the application of advanced automation or artificial intelligence techniques, a few basic steps that can be more practically implemented should be taken. These include: 1) get more information available on-line in a networked environment, 2) standardize information in terms of media, structure, and content, 3) begin requiring submission of documents in electronic form. Efforts are already being made to do these, an example being the TMIS database [4].

The long-term goal of this research is to develop approaches for the integration of knowledge engineering with the traditional engineering activities. The short term goals have been to identify potential knowledge sources and to identify opportunities for integration.

## 9. References


[1] Advanced X-Ray Astrophysics Facility Request for Proposal, RFP No. 8-1-0PP-01663-54, 4 January 1988. George C. Marshall Space Flight Center, Marshall Space Flight Center, Alabama 35812.

[2] Hubble Space Telescope Command List, ST/DM-01, SDM-1001B, 25 April 1986, Lockeed Missiles & Space Company, Inc., Space Systems Division, Sunnyvale, CA.

[3] Hubble Space Telescope Design Engineering Knowledgebase Research Progress Report 1.

[4] Hubble Space Telescope Design Engineering Knowledgebase Research Progress Report 2.