Semantics for Digital Engineering Archives Supporting Engineering Design Education

William Regli, Joseph B. Kopena, Michael Grauer, Timothy Simpson, Robert Stone, Kemper Lewis, Matt Bohm, David Wilkie, Martin Piecyk, and Jordan Osecki

■ *This article introduces the challenge of digital preservation in the area of engineering design and manufacturing and presents a methodology to apply knowledge representation and semantic techniques to develop digital engineering archives. This work is part of an ongoing, multiuniversity effort to create cyber infrastructure–based engineering repositories for undergraduates (CIBER-U) to support engineering design education. The technical approach is to use knowledge representation techniques to create formal models of engineering data elements, work flows, and processes. With these techniques formal engineering knowledge and processes can be captured and preserved with some guarantee of long-term interpretability. The article presents examples of how the techniques can be used to encode specific engineering information packages and work flows. These techniques are being integrated into a semantic wiki that supports the CIBER-U engineering education activities across nine universities and involving more than 3500 students since 2006.*

Digital preservation is the mitigation of the deleterious effects of technology obsolescence, media degradation, and fading human memory (Gladney 2006). One of the fundamental challenges facing those developing digital preservation solutions is the development of digital representations that are self-describing and assured to be interpretable over the long lifespans required by archival applications.

This article utilizes semantic web technologies to create self-describing representations for archiving engineering data. The overall approach is to design a set of extensible ontologies that describe engineering file formats and their underlying data models, along with the software and hardware tools used to create and transform this data. Further, these form the basis for representations of process work flows and aggregations of engineering objects that capture relationships among the files and key data transformation processes. The ontologies can also be used to record the relationships of a file to different versions of itself as it evolves over time and can record provenance metadata about a file such as the creating agent, time, and location. This allows for digital archives that can record knowledge about their contents rather than merely store data.

There are two specific contributions presented in this article. First, we develop a formal model for an extensible format registry for engineering data elements and software. The format registry contains the fundamental ontology with which engineering data and processes can be captured and preserved with some guarantee of long-term readability. Second, we present a

 ISSN 0738-4602

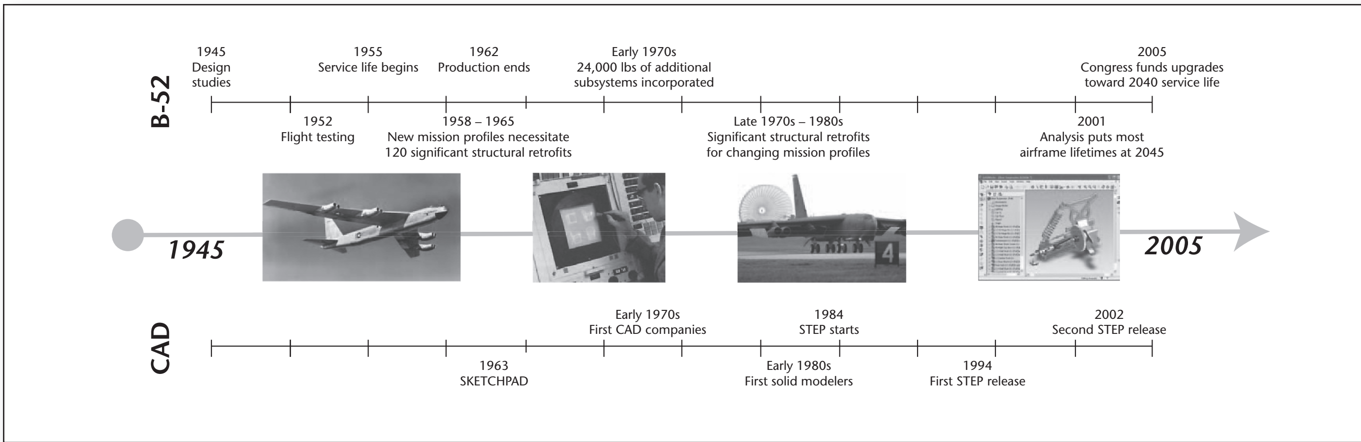


Figure 1. Life Cyle of the B-52 Versus the History of CAD and Digital Engineering Artifacts.

method of capturing and representing engineering work flows based on the format registry using the Process Specification Language (PSL) (Grüninger and Menzel 2003).

This offers a new approach to creating long-term digital archives. Prior work in digital preservation falls into two categories: conversion, where digital files are constantly updated and their encodings translated; and emulation, where the original execution environment will be emulated on future platforms. The semantics-based approach in this article aims to support design knowledge capture (Regli et al. 2001) into well-defined neutral forms, enabling future users the ability to extract meaningful information from archived digital objects.

The work presented in this article is currently supporting the cyber infrastructure–based engineering repositories for undergraduates (CIBER-U[1]) collaboratory (Simpson et al. 2007), an initiative to improve the ability of educators to teach engineering design by developing an extensible library of virtual product dissections. Since its inception in 2006, this collaboratory has been used by more than 3500 students at nine universities.

The Challenge of Digital Engineering Archives

The relationships among shape and form, structure and function, and behavior and semantics are among the most fundamental questions studied by science and engineering—and it is precisely these relationships that must be captured and preserved in digital engineering archives. For many industries (aerospace, civil engineering and architecture, shipbuilding, geoengineering), engineering design and manufacturing knowledge needs to be preserved over 50- to 75-year lifespans (Thilmany 2005).

For example, the Boeing B-52 Stratofortress (figure 1) was conceptually designed in the late 1940s, first deployed in 1955, and planned to be in operation until 2040.[2] This aircraft's lifetime extends not just over changes in CAD software packages, but across the development of the entire computing and software industry itself. To maintain the B-52, originally drawings were converted to digital files—files that must be continually imported into subsequent new versions of CAD software (usually vendors only support the most recent version). Any data that needs to be usable in the future must have constant "data hygiene" practices performed on it, updating it as standards change and maintaining any relationships it has with other files.

Traditional digital data management in such a situation is highly dependent on the proprietary formats of commercial software systems, proving it hard to guarantee the utility of data over long periods. Hence, while nearly all modern engineering domains view the three-dimensional computer-aided design (CAD) model as indispensable, the engineering part print (that is, blueprint or two-dimensional drawing on paper, aperture cards, microfiche) remains as the principal method of design knowledge archival. From an archival standpoint, much of the knowledge generated during the modern engineering enterprise (that is, by three-dimensional CAD, simulation, and so on) is simply lost. Even if the CAD files are archived, the supporting infrastructure required to access and understand these designs will be obsolete and unusable. The rich digital knowledge in three-dimensional CAD about features, design and production work flow, manufacturing processes, and artifact behavior are simply not captured or are lost in a myriad of data translations.

Other digital preservation domains (such as audio, video, still image, and so on), have developed format registries in order to better capture archival issues (Abrams and Seaman 2003; U.S. Library of Congress 2006).[3] Such efforts are largely static and focused on relatively few media types. In contrast, a comprehensive representation of a single engineering artifact might encompass thousands of files in hundreds of different formats.

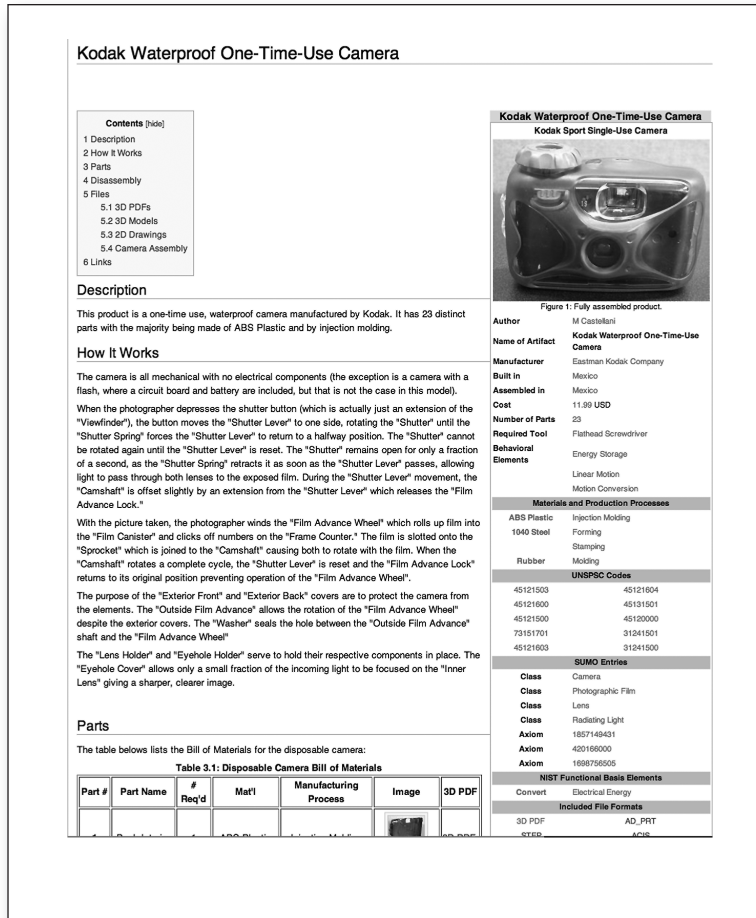
CIBER-U

CIBER-U (Simpson et al. 2007) is an ongoing collaboration among the engineering faculties of nine universities (Pennsylvania State University; State University of New York, Buffalo; Drexel University; Missouri University of Science and Technology; Virgina Polytechnic Institute and State University; Bucknell College; Sweet Briar State University; Norfolk State University; and Northwestern University) to create a national product dissection collaboratory to support engineering education. Product dissection has been used successfully in a variety of ways to actively engage students in learning engineering design (Sheppard 1992). Dissection can be used to increase awareness of the design process (Otto and Wood 2006) as well as teach competitive assessment and benchmarking (Marchese et al. 2003).

Despite the numerous advantages of using product dissection throughout the engineering curricula, product dissection has not yet become a national model for engineering design education. Products, tools, and their upkeep can be costly, workspace and storage space can be difficult to obtain, and even the best crafted dissection assignments can end in chaos. CIBER-U aims to create a living repository of product dissection activities, suitable for all levels of engineering undergraduate curricula. With CIBER-U, instructors use and contribute to an archival corpus of digital design repositories that include CAD models, simulation data (including kinematics, dynamics, physics, and so on), video, and other multimedia. The long-term goal is for this corpus to become a key element of engineering education nationally.

Example of CIBER-U Content. The principle contents of CIBER-U are "case studies," an example of which is shown in figure 2 and may include (1) a brief description of the product and how it works; (2) a list of all its parts (such as a bill of materials), including the quantity, material, manufacturing process, and a photograph of each part; (3) step-by-step product disassembly instructions; (4) a set of files including three-dimensional CAD models and two-dimensional drawings of each part and an assembly model of the entire product; and (5) descriptions of the functional, behavioral, and energy interactions of the project components and how they contribute to achieve the overall design objectives.

Some case studies also include animations (in CAD) of their disassembly and videotaped presentations of them being manually disassembled. These are particularly useful for first-time instructors who may not be familiar with a particular product, especially if they have never dissected it before.



Kodak Waterproof One-Time-Use Camera

Contents [hide]
1 Description
2 How It Works
3 Parts
4 Disassembly
5 Files
5.1 3D PDFs
5.2 3D Models
5.3 2D Drawings
5.4 Camera Assembly
6 Links

Description

This product is a one-time use, waterproof camera manufactured by Kodak. It has 23 distinct parts with the majority being made of ABS Plastic and by injection molding.

How It Works

The camera is all mechanical with no electrical components (the exception is a camera with a flash, where a circuit board and battery are included, but that is not the case in this model).

When the photographer depresses the shutter button (which is actually just an extension of the "Viewfinder"), the button moves the "Shutter Lever" to one side, rotating the "Shutter" until the "Shutter Spring" forces the "Shutter Lever" to return to a halfway position. The "Shutter" cannot be rotated again until the "Shutter Lever" is reset. The "Shutter" remains open for only a fraction of a second, as the "Shutter Spring" retracts it as soon as the "Shutter Lever" passes, allowing light to pass through both lenses to the exposed film. During the "Shutter Lever" movement, the "Camshaft" is offset slightly by an extension from the "Shutter Lever" which releases the "Film Advance Lock."

With the picture taken, the photographer winds the "Film Advance Wheel" which rolls up film into the "Film Canister" and clicks off numbers on the "Frame Counter." The film is slotted onto the "Sprocket" which is joined to the "Camshaft" causing both to rotate with the film. When the "Camshaft" rotates a complete cycle, the "Shutter Lever" is reset and the "Film Advance Lock" returns to its original position preventing operation of the "Film Advance Wheel".

The purpose of the "Exterior Front" and "Exterior Back" covers are to protect the camera from the elements. The "Outside Film Advance" allows the rotation of the "Film Advance Wheel" despite the exterior covers. The "Washer" seals the hole between the "Outside Film Advance" shaft and the "Film Advance Wheel"

The "Lens Holder" and "Eyehole Holder" serve to hold their respective components in place. The "Eyehole Cover" allows only a small fraction of the incoming light to be focused on the "Inner Lens" giving a sharper, clearer image.

Parts

The table belows lists the Bill of Materials for the disposable camera:

Table 3.1: Disposable Camera Bill of Materials

Part #	Part Name	# Req'd	Mat'l	Manufacturing Process	Image	3D PDF

Kodak Waterproof One-Time-Use Camera	
Kodak Sport Single-Use Camera	
Figure 1: Fully assembled product.	
Author	M Castellani
Name of Artifact	Kodak Waterproof One-Time-Use Camera
Manufacturer	Eastman Kodak Company
Built in	Mexico
Assembled in	Mexico
Cost	11.99 USD
Number of Parts	23
Required Tool	Flathead Screwdriver
Behavioral Elements	Energy Storage
	Linear Motion
	Motion Conversion
Materials and Production Processes	
ABS Plastic	Injection Molding
1040 Steel	Forming
	Stamping
Rubber	Molding
UNSPSC Codes	
45121503	45121604
45121600	45131501
45121500	45120000
73151701	31241501
45121603	31241500
SUMO Entries	
Class	Camera
Class	Photographic Film
Class	Lens
Class	Radiating Light
Axiom	1857149431
Axiom	420166000
Axiom	1698756505
NIST Functional Basis Elements	
Convert	Electrical Energy
Included File Formats	
3D PDF	AD_PRT

Figure 2. CIBER-U Wiki Page for Kodak One-Time-Use Camera Product Dissection Case Study.

Use of CIBER-U materials is not a "one-time" event. Having a shared repository allows educators to expand the range of things that can be accomplished in a course. For example, in one of the CIBER case studies of an internal combustion engine[4] the initial course offering focused on developing CAD models for subsystems in an engine and documenting how each subsystem works; the second course integrated these CAD models to create models of the overall system and document overall how the engine works; and the third course created animations of how each system works based on these archived CAD models.

CIBER-U Archiving Challenges. CIBER-U is an exemplar for the wider challenges in the creation of digital engineering archives. Consider the following:

> CIBER-U data is highly heterogeneous. Different CAD or CAE systems are used across institutions, resulting in many different file formats populating the case studies.

Dissection studies involve a number of different activity work flows to interpret and document the product.

For some case studies, product life-cycle data includes knowledge from different domains and software systems (such as physics-based models, kinematics, dynamics, and so on).

Data files for engineering models can be large (that is, hundreds of files and gigabytes are required to represent even simple parts) and have complex internal structures (such as geometry, math, physics, and so on).

Most importantly, for CIBER-U materials to be of ongoing usefulness, they must be interpretable by future engineering students and CAD or CAE systems.

Viewed in this way, CIBER-U is similar to problems in the management of large-scale scientific data sets. However, the engineering domain offers significant complexity due to the size and internal structure of engineering and CAD files. This requires a practical approach to capturing engineering knowledge at the time of creation and to transforming this knowledge into format structures that can increase the usefulness of this data over time.

Technical Approach

The approach to CIBER-U's infrastructure is based on combining wikis and simple knowledge and data-capture tools with formal knowledge representation and reasoning based on open standards. Semantic web technologies in particular are applied to structure engineering knowledge and data in a manner reasonably assured to be persistent and consistently interpretable over long periods of time. Primary elements of this work include the following:

We define and extend ontologies to represent engineering design and process knowledge in information packages that collect and describe digital data associated with engineering artifacts. These information packages organize the files and objects created, providing a framework for associated domain knowledge and metadata.

We create a knowledge base of software engineering applications and related file formats to construct an extensible, ontology-based format registry. This enables specification of relationships among information packages and files, associates file types with engineering activities, and supports reasoning for system tasks.

We develop and apply ontologies of domain knowledge such as artifact metadata and engineering process work flows. Artifact metadata includes relationships between parts and assemblies, notions of function, materials, and other documentation. Workflows in this setting are similar to those in business processes reengineering and scientific areas (Gil et al. 2007). Capturing them and their context requires knowledge of process models, engineering activities such as design decision and analysis, applicable tools, and electronic or physical work environments.

Development of CIBER-U's infrastructure is guided by four key tenets derived from our experiences with both design repositories and engineering education:

Capture everything. Over a long enough time frame, it is hard to predict what data will be useful and what will not. In addition, as the cliche goes, disk space is cheap. The archive should then preserve as much as possible, without worrying too much about whether any particular item will be useful in the future. Further, rather than requiring perfect a priori representations of the data, the goal should be to structure data so as to reduce the digital archeology required to utilize it, and accept that some amount of such work may always be necessary.

Intrude as little as possible. Archival systems have very limited ability to deploy new interfaces and software, add bookkeeping, and otherwise change engineers' processes without pushing them to ignore the system. This is particularly true in CIBER-U's educational setting, in which users are generally relatively novice computer users and are focused on the engineering tasks at hand rather than archival concerns. In an ideal world an archive system should be transparent to users, augmenting their capabilities without imposing additional constraints or effort.

Accept informality, within formal structure. We cannot truly capture all of the semantics in the engineering domain, for both theoretical and practical reasons. As an example, it's entirely plausible that some data might require notions of transitive closure or connectedness of a graph. Neither of these is expressible even in first-order logic, and moving to yet more expressive frameworks presents its own myriad challenges. The goal must be to constrain interpretation and extend the representation over time rather than solely accept a perfect a priori formalization. In some sense, we aim to formalize the structure of the data and define the domain semantics as much as possible when required, without necessarily requiring complete domain formalization.

Apply and adapt reasoning as necessary. Similarly, we cannot and need not always apply the most powerful reasoning possible. It is often more practical to leverage semantics, structure, and syntax of the data to enable more specialized and simplistic reasoning. Formalization defines interpretation of the data and the inferences that may be made, enabling that reasoning to be well defined and demonstrably correct. That does not necessarily

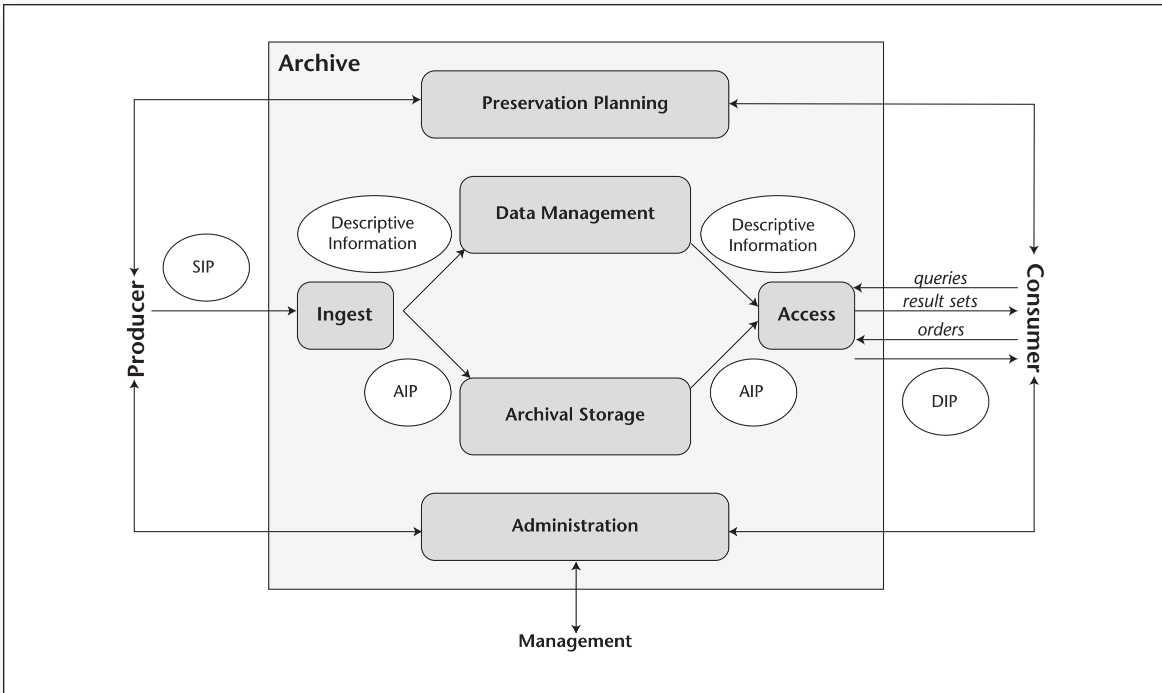


Figure 3. Core Components and Flows in the ISO 14721: 2003 Open Archival Information System (OAIS) Reference Model.

require that full, general logical entailment to match the expressiveness of the representation be applied in all tasks.

The following sections illustrate these concepts and provide examples illustrating how they have been applied to complex engineering data elements archived in CIBER-U.

OAIS Reference Model

Like much work in digital archiving, CIBER-U is framed in terms of ISO Standard 14721:2003—Reference Model for an Open Archival Information System (OAIS) (Consultative Committee for Space DataSystems 2002). The OAIS reference model provides a common language of archival systems, in the form of an abstract breakdown of the information flows and functional components. Figure 3 presents the core structure of the OAIS model, of which the four components of interest here are as follows: (1) *Ingest* provides an automated or manual interface through which data is submitted to be archived. (2) *Archival storage* implements functionality for permanent storage of the data, including tasks such as compression, media migration, error checking, and disaster recovery. (3) *Data management* implements indexing and search functionality to identify, navigate, and query archival content. (4) *Access* provides an interface to query and retrieve content. Data management is used to identify content to be retrieved, which is then requested of archival storage.

Each of these components operates on information packages, which bundle data, information on the syntax and semantics of that data, and preservation description information used to identify, verify, and certify the package. As shown in figure 3 these are sometimes identified as information packages for submission (SIP), archive (AIP), descriptive, and dissemination (DIP), according to their role.

Notably, OAIS is a reference model, not an architecture. Implementations contain their own set of human and machine components and procedures that may not directly resemble these components, but do map to the OAIS model.

Wiki for Engineering Archives

In CIBER-U the archive's ingest and access components as well as its information packages are commingled into a single element: a wiki. Figure 2 presents a sample page developed by students in a product dissection course, documenting a camera.

Wikis effectively meet our requirement to intrude as little as possible into engineers' processes. In CIBER-U, students expect to and are already tasked with preparing reports of their studies. Mandating that the reports be developed within a wiki imposes a constraint, but in return provides for easy collaboration among group members, including version control. Many students are also comfortable working with and quickly learn wiki interfaces and markup.

Figure 4 charts some of the elements generated in even a simple study such as that of figure 2.

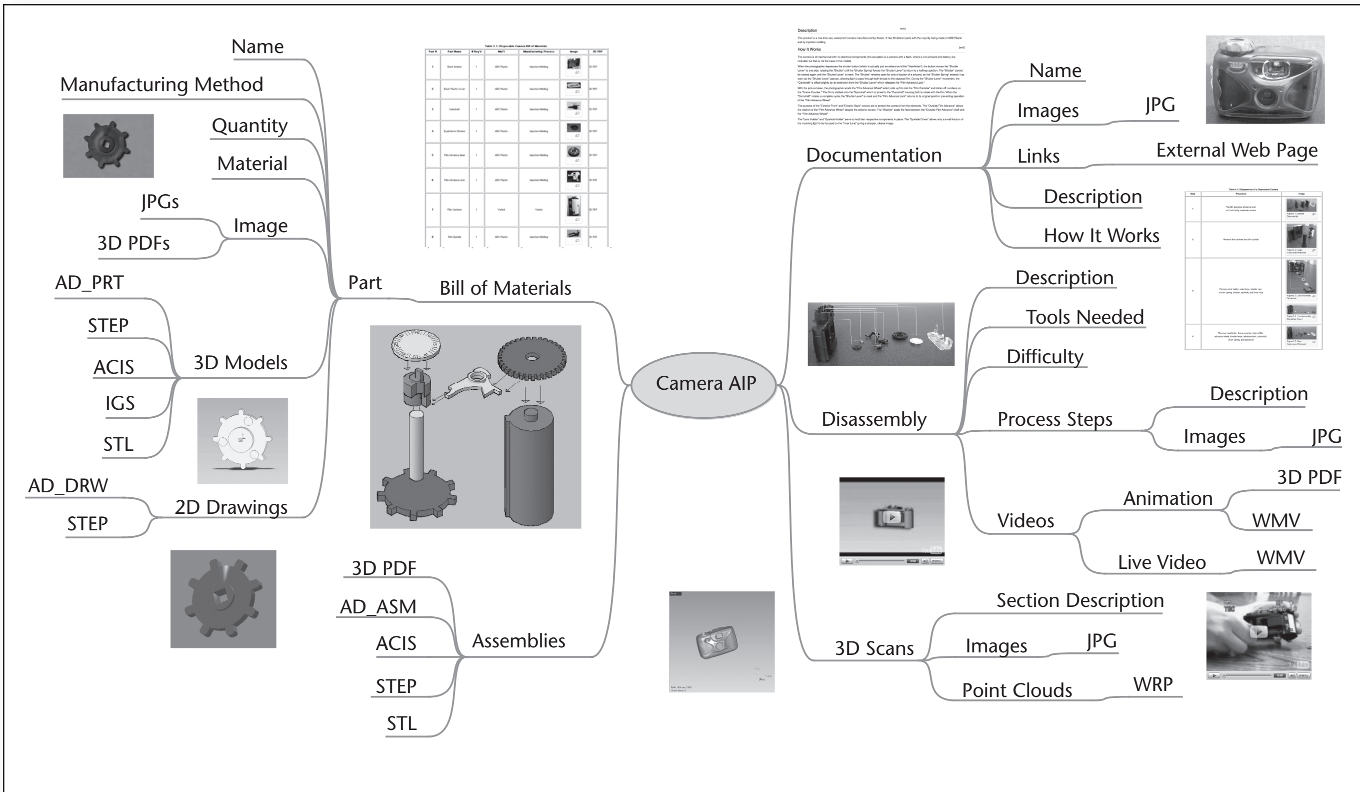


Figure 4. Information Package Elements for Camera Study.

```
{{overview-images |=
{{image| view=Front | src=DSCN0166.JPG }}
}}

{{start-bom}}
{{bom-item | partid=1 | name=Back Interior |=
  | material=ABS Plastic |=
  | process=Injection Molding |=
  | image=Camera back interior.JPG |=
  | model=back interior.pdf | modelType=3D PDF}}
{{bom-item | partid=17 | name=Shutter |=
  | material=1040 Steel | process=Stamping |=
  | image=Camera Shutter.JPG | imageType=JPEG}}
{{end-bom}}
```

Figure 5. Wiki Markup for Data Elements in Camera Study.

CIBER-U's wiki pages form the information packages that collect and map all of this information. In our courses and studies so far this has included three-dimensional solid models at multiple resolutions or levels of detail; derivative models for specific analyses, for example, surface meshes for finite element analysis (FEA); documentation of process history and rationale, accommodating both actual design changes and information operations, such as creation of those derivative models; photographs and three-dimensional scans; disassembly videos; material listings; and simulation code.

The challenge then is capturing knowledge about these elements and the relationships between them in a structured fashion for machine use, without introducing unfamiliar languages or additional tasks for the engineering users. CIBER-U accomplishes this in its wiki-based information packages through a combination of semantic MediaWiki (Krötzsch et al. 2007) and custom template markup. A snippet of such markup from a camera study is given in figure 5.

CIBER-U currently relies on a number of custom templates and processors for extracting information from its information package wiki pages, for two reasons: (1) specialized domain and task-specific templates make it easier for untrained users to capture domain knowledge in a structured scheme, as in figure 5; and (2) a semantic MediaWiki notation may only make an assertion about the object associated with its host page, making it cumbersome to add small amounts of statements about related objects, such as in a bill of materials.

Ad hoc processing to address these points does introduce some fragility into the system, as the associated microlanguages and scripts may themselves decay. Current work in CIBER-U aims to mitigate this through a formal framework and implementation mechanism that ties together background ontologies, custom templates, and information extraction in a more general and robustly maintainable fashion.

Regardless, an information package is extracted from the wiki pages and encoded as instance data within the Ontology Web Language (OWL)/ Resource Description Framework (RDF)4. An example of this from the snippet in figure 5 is presented in figure 6. The ontologies that structure that representation and examples of the reasoning that may be performed over it are discussed in the following sections.

Implementing Engineering Ontologies

While there has been considerable research on ontologies for engineering design processes, none has particularly addressed the taxonomy of engineering file formats, their interactions, or preservation needs. Rather than focusing on abstract aspects of design, the needs of CIBER-U require emphasis on files as atomic elements in the system and ontologies. After all, files and their contents are the objects that the archive is preserving over time and needs to reason about.

A map of ontologies used and under development in CIBER-U is shown in figure 7; it divides into three areas: (1) the core engineering information package structure that organizes and connects files and metadata; (2) A format registry providing a knowledge base of file formats, applications, and associated domain knowledge; and (3) contextual knowledge about the information package and its data, such as function or design work flows.

Each of these ontologies has been developed under a methodology based on competency questions (Grüninger and Fox 1995). A core principle in this approach is the simple but important notion that the adequacy of an ontology is defined in terms of its ability to support specific application reasoning and query requirements, as opposed to more subjective or philosophical gauges of its completeness. In turn, those application requirements may be defined through prototypical queries the ontology must be able to answer. Examples from the CIBER-U ontologies include: What models are associated with a given artifact? What artifacts have associated photographs? What installed software tool can translate between a given pair of solid model formats? What analysis steps lead to the creation of a given model?


```
<part:Assembly rdf:about="&wiki;KodakOneTimeCamera#">
  <ip:image>
    <fr:JPEG rdf:about="&wikid;DSCN0166.JPG" />
  </ip:image>

  <ip:hasBillOfMaterials>
    <ip:BillOfMaterials>
      <ip:BOMEntry>
        <ip:partID>1</ip:partID>
        <ip:entry>
          <part:Part rdf:about="&wiki;BackInterior#">
            <part:material rdf:resource="&mat;#ABSPlastic" />
            <part:manufacturingProcess
              rdf:resource="&manf;#InjectionMolding" />

            <ip:hasImage>
              <ip:Image rdf:about="&wikid;Camera back interior.JPG">
                <fr:hasFormat rdf:resource="&fr;JPEG" />
              </ip:Image>
            </ip:hasImage>

            <ip:hasShapeModel>
              <ip:SolidModel rdf:about="&wikid;back interior.pdf">
                <fr:hasFormat rdf:resource="&fr;PDF3D" />
              </ip:SolidModel>
            </ip:hasShapeModel>
          </part:Part>
        </ip:entry>
      </ip:BOMEntry>


      <ip:BOMEntry>
        <ip:partID>17</ip:partID>
        <ip:entry>
          <part:Part rdf:about="&wiki;Shutter#">
            <part:material rdf:resource="&mat;#Steel1040" />
            <part:manufacturingProcess
              rdf:resource="&manf;#Stamping" />

            <ip:hasImage>
              <ip:Image rdf:about="&wikid;Camera Shutter.JPG">
                <fr:hasFormat rdf:resource="&fr;JPEG" />
              </ip:Image>
            </ip:hasImage>
          </part:Part>
        </ip:entry>
      </ip:BOMEntry>
    </ip:BillOfMaterials>
  </ip:hasBillOfMaterials>
</part:Assembly>
```


Figure 6. OWL/RDF Data Extracted from CIBER-U Wiki.

The following three subsections describe each of these areas and their interconnections in more detail.

Engineering Information Packages

Central to the CIBER-U ontologies is the definition of engineering information packages. Following the OAIS model, information packages may be divided into five components: *Data:* The actual content to be archived. *Context:* Structure and meaning of the data. *Provenance:* History and rights

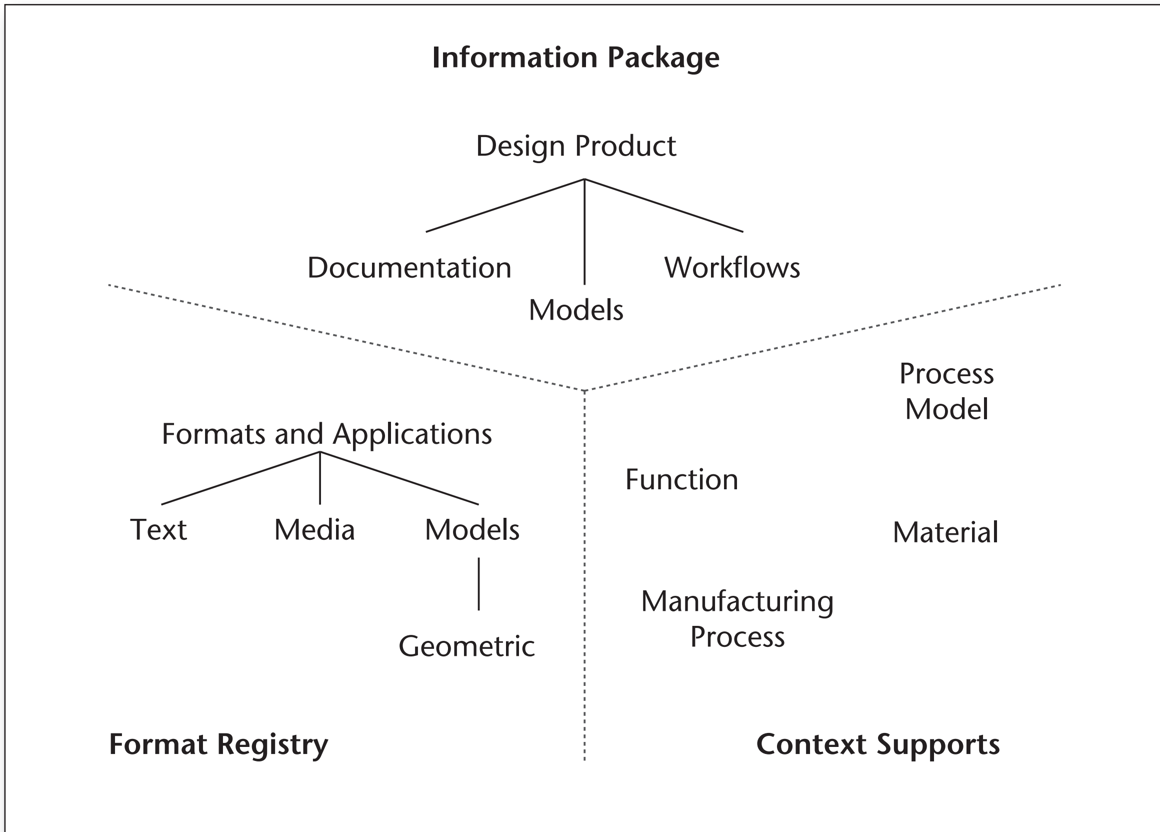


Figure 7. Ontologies used in CIBER-U.

for the data. *Reference:* Identifiers for the data. *Fixity:* Keys and checksums used to check for integrity.

In CIBER-U the data component is all of the digital files associated with the information package. Reference and fixity are the URL associated with the information package wiki page and checksums for the wiki page and data files. Context and provenance are the structured information extracted from the wiki markup and page metadata, describing how all of the objects in the data are related.

The root concept of the context information is the design product, which is the core of the information package. To summarize some key elements of the ontology structure: Design products may be either parts or assemblies and may be associated with documentation, models, and work flows. Each of these is in turn the root of a taxonomy of data object types. For example, documentation includes requirements, specifications, design records, videos, images, reports, engineering change requests, user manuals, and bills of materials. Model objects includes geometric (CAD three-dimensional solid and two-dimensional models), shape (mesh and point cloud data), tolerance (manufacturing precision requirements), kinematics, dynamics, and behavior (models of part motions in assemblies), function (the purpose of each component), and materials. Workflows comprise design, analysis, assembly, fabrication, translation, reverse engineering, simulation, manufacturing, and inspection. All files related to a product are included in the information package architecture in some way through an ontology instance describing the digital file and connected to the design product through one of these concepts.

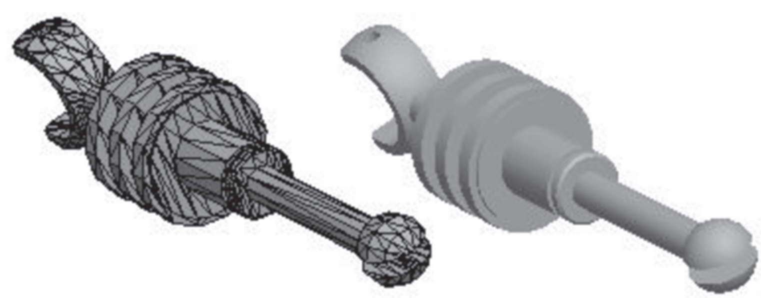
CIBER-U's information package also includes some support for capturing provenance—the history, users, and access rights for a file. A small ontology based on abstract requirements for engineering provenance data (Ram and Liu 2006) and OWL Time[6] is used to encode information about who changed a file or information package, how, and when.

Engineering Format Registry

An example of a preservation resource is CIBER-U's format registry. In other archiving contexts, a format registry is used to capture information about how files were and may be used in the engineering process, how they may be created or modified, and other information associated with part and assem-

Product Data Type	Traditional Data Format	Web-Enabled Data Format
3D solid model	STEP	VRML
2D engineering drawings	DXF, DWG, and others	DWF
Images	TIFF, GIF, JPEG, and others	GIF, JPEG
Unformatted documents	TXT	XML, HTML, TXT
Formatted documents	MS Word, PostScript	PDF, MS Word
Forms	Lotus 123, MS Excel	HTML
Sectors of database	Database	XML, HTML
Audio	WAV, and others	MP3
Video	MPEG, and others	MOV
Animations	—	VRML, Flash

A Bestiary of Engineering Formats



A Simple Cam in Both Solid and Mesh Format

Figure 8. Two Challenges with Engineering Objects: The Diversity of Related File Elements and Their Internal Representational Complexities.

bly models, design documentation, simulations, and other files. For CIBER-U, we extended the core information package ontology as an ontology and knowledge base of file formats and applications to create a small ontology of file format properties and a large knowledge base of ontology instances. Each instance contains the particular property values for the format it represents. Metadata properties in the ontology have largely been drawn from several similar format registry efforts (Abrams and Seaman 2003; U.S. Library of Congress 2006).[3] One notable simple property in the ontology relates formats to their common file-name extensions, enabling archive components to reason about formats a file might be in if it has no explicit type.

As an ontological issue, file formats could be viewed as classes of file objects and their defining properties. For CIBER-U this approach seemed cumbersome for our applications; describing formats as instances to which files point was deemed more intuitive and practical. A small extension in the future will rarefy this relationship, with explicit format instance objects to capture instantiated parameters.

To foster collaborative development, extension, and maintenance, the format registry knowledge base is implemented as a collection of semantic MediaWiki pages. Each page describes a particular format or application using a standard template, and semantic MediaWiki's export tools generate RDF for the format object. Similar to the wiki approach taken to capturing information packages, this enables domain users without significant training in knowledge representation to participate in developing the knowledge base.

The Hierarchy of Engineering Formats. More novel and arguably more useful for the archive than metadata properties of formats is their role within the engineering process. This is captured in CIBER-U by placing each format as an instance within a hierarchy mirroring the information package map of engineering data described above. This provides for denoting the design activities formats may potentially be used in. As an example, JPEG is an instance of both lossy image format and raster image format, which subclass image format, which in turn is a subclass of documentation format.

Challenges Specific to CAD/CAE Data. A large portion of the taxonomy for CIBER-U focuses on geometric models, capturing knowledge of different shape representations and their representational power and attributes. Even describing the simplest engineering object may require capturing both a diverse array of file types and also their interactions, interdependencies, and respective provenance. Many of these classes are associated with well defined, mathematical expressions of their properties and capabilities. However, tying these definitions together into a single, practically useful computational ontology would be a significant challenge.

Consider the table (Zhang, Shen, and Ghenniwa 2004) of file types necessary for CIBER-U shown in figure 8a; for large engineering enterprises a comprehensive taxonomy of file types of interest may be much more vast. For example, simply having the CAD model for a design may not be enough to make sense of the object. If you need to remanufacture the objects, you may also need to have the design requirements documents as well as models for the in-process shapes that would reveal the process plan. This information is not resident in any single file; rather it is present in the network of relationships among the files and elements of their internal structure.

Further complicating matters is the inherent complexity of the internal structures of engineering data objects. In the course of the engineering process many derivative model representations must be created for the various work flows central to design, manufacturing, and life-cycle activities. This process is affected by several fundamental (and thorny) theoretical issues in computational topology and the theory of computer-aided design: the representations in our computers (or in these files) are really descriptors for a class of mathematically ideal objects that can be realized within the mathematical tolerances of the data structures describing the nominal geometry and topology of a computer system. One can have exactly the same object represented in Pro/ENGINEER as a collection of NURBS surfaces, as a triangle mesh, or as a set of analytic surfaces as an ISO 103033 STEP file: vastly different underlying representations for the same object (as shown in Figure 8b). Further complicating matters, one cannot simply translate the files into some ideal format without significant knowledge loss because fundamental properties can change in the process; that is, a watertight model can become nonmanifold, triangles that previously did not interpenetrate now overlap, and so on.

The approach used in CIBER-U is that the best defense against knowledge loss is maintaining multiple, perhaps redundant, representations of objects. Further, rather than duplicate work being performed by the international standards community, we capture the attributes of different shape representations and their representational power.

Applications. CIBER-U's format registry also captures knowledge about applications and their relationships to file formats. Similarly to file formats, a small ontology of application properties is defined, using which a large collection of ontology instances is maintained through a semantic MediaWiki. Most interesting of the current application object properties are the application class and their inputs and outputs. Application class echoes the taxonomy of file format types and roles, discussed above. Inputs and outputs relate an application to the kinds of file formats it may read and write. This knowledge enables the archive to reason about what transformations may be performed on a given data set, verify the feasibility of a captured work flow, and other tasks.

Context and Process Supports

The final set of components in CIBER-U's suite of ontologies directly fill in knowledge about different elements in the information package. For example, in previous work we have developed semantic web representations of artifact function (Kopena and Regli 2003) that are easily incorporated into the function element of the information package. This can provide significant additional metadata about artifacts that may be reasoned on to support search, classification, and other archive functionality. Future work in CIBER-U includes developing or adapting additional supporting ontologies for materials (Ashino and Fujita 2006), manufacturing processes (Lemaignan et al. 2006), requirements documents (Lin, Fox, and Bilgic 1996), and design (Darlington and Culley 2008) that may be integrated into the information package.

Processes and Workflows. One contextual element currently supported in CIBER-U's ontologies are generic processes, which have been used to model disassembly and assembly as well as simulation and analysis work flows. At the top level of this ontology are process objects, which have associated assumptions, constraints, and parameters. Simple processes of sequences and branches are modeled by relating process objects to collections of activities and other processes. Activities are further decomposed into generic activities of transformation and analysis. Both have inputs and outputs that may be instances of design products or digital files, discussed previously.

As an example, one CIBER-U case study, in figure

9, has explored more advanced features by archiving the design process of a novel robot design. The design, in figure 9, has a number of features compared to other snake-inspired designs, such as linear motion, and requires new types of control algorithms. Complex designs such as this one are complex to control, and one method researchers are using to cope with this complexity is to design control algorithms using simulation and analysis through a physics-based virtual model.

Figure 9 shows the basic process model that was developed to archive the simulation and analysis work flow. In this process model, the work flow is captured as a set of dependencies based on the format registry ontology elements. For example, design data, in the form of the CAD solid models of the robot, is in specific file format instances (information package, of Pro/Engineer .prt and .asm files). To create the simulation of the robot, other derivative file formats must be created to provide the inputs to the simulation; these must also be associated with the original model's information package. The translation task process model relates the the different model representations (such as solid, mesh, and so on) that capture the robot's structure, contacts, densities, coefficients of friction, and geometric axes needed for positioning. Using a simulation application (in this case, the Open Dynamics Engine), a designer can develop a code for a gait and test its effectiveness through a physics-based simulation.

Deployment, Assessment, and Future Work

To date, more than 5,000 students have participated in one or more CIBER-U activities at any of the nearly dozen institutions using the repository. These students range from high school students working as part of summer research experience programs to undergraduates as well as graduate students. These activities developed product dissection exercises to be shared and used across all of the partner institutions and created rubrics and evaluation methods to assess these activities. CIBER-U data sets and exercises have been being actively incorporated into curriculum materials at the involved institutions since 2006, and the team is engaged in longitudinal studies to assess the effectiveness of these materials.

In this context, the use of semantic representations is largely invisible to the end users. The format registry, work flows, and rule sets are operational behind the semantic MediaWiki interface used by the students and faculty. The information package representations, work flows, context models, and so on all facilitate the back-end processing of data elements uploaded into the CIBER-U semantic wiki. For example, the format registry drives the automated processing of model translations into neutral file formats and the indexing of these three-dimensional models for queries. Context information provides associations among file types based on date and work flow relationships (information package models can be automatically connected with associated simulation or manufacturing data). Assessing the effectiveness of the semantic approach cannot be done directly; given the semantic technology's functional purpose is to improve knowledge retention and data viability over time, assessment will be done longitudinally.

This noted, the CIBER-U team has assessed student performance through their use and exposure to our cyber-infrastructure engineering data sets. One assessment by SUNY-Buffalo compared the improvement in test scores of the CIBER-U students with students who did not use the CIBER-U semantic wiki repository for the first and final exams in the 2008 academic year. The CIBER-U students improved an average of 14.1 points and the non-CIBER-U students gained 8.8 points (significant at $p < 0.05$). This increase may be for various reasons; the team believes that exposure to the repository information and representations played an important role in the CIBER-U students' learning and comprehension. Inasmuch as the semantic structures behind CIBER-U enhance the educational delivery and ease of maintenance, the use of AI techniques has been a great benefit to the CIBER-U system developers.

In the future, the team sees CIBER-U enabling "virtual dissection"—using the data to teach students about disassembly without having the physical product in hand (for example, like a virtual frog or the visible human). While physical dissection (with only products and tools) and virtual dissection (with only a computer screen and haptic interfaces) represent different ends of the dissection spectrum, most CIBER-U activities have to date been really only "cyber-enhanced" dissection, with some aspects of both physical and virtual. The data and associated semantic structures are vital to the creation of a richer data set for both virtual and cyber-enhanced dissection domains. Eventually, the CIBER-U repository will include larger products, those that would not typically be available for dissection in a classroom due to cost or size (for example, refrigerator, car, aircraft, and so on).

In addition to the information archived on the CIBER-U semantic wiki, the team has two design repositories that archive CAD data, the National Design Repository[7] and the Oregon State University (OSU) Design Repository, capturing function and behavior models for electro-mechanical artifacts.[8] The National Design Repository, an archive of more than 55,000 CAD models, designs, and drawings, spans a variety of domains (Regli and Gaines 1997). Over the past decade, the repository

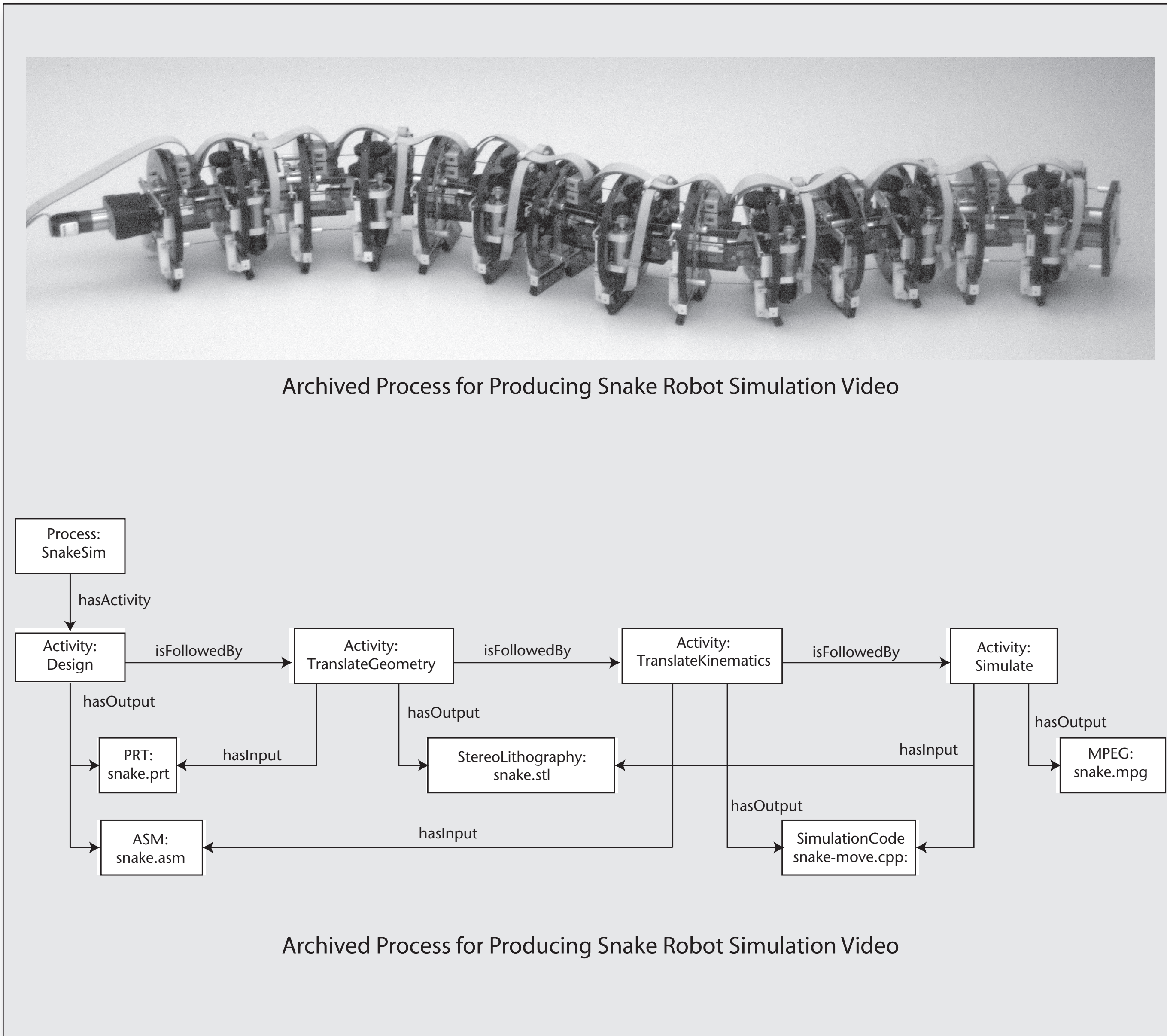


Figure 9. Case Study in Archiving Design of Biologically Inspired Robot.

has been considered the canonical data set for work in intelligent CAD, shape matching, process planning, and data translation. The OSU Design Repository serves as a hub for designers for information exchange and design-generation tools. The infrastructure supporting these two applications is the design repository information ontology (Bohm et al. 2008), currently being integrated with the CIBER-U wiki ontologies. This information ontology describes what types of design information can be stored, the relationship of those elements, and extensibility for including new and additional types of design information.

Conclusions

This article presented a methodology to support long-term preservation of "born-digital" engineering artifacts. Digital preservation for engineering enterprises poses unique challenges due to the complexity and diversity of the datatypes involved in CAD or CAM, simulation, and other aspects of the product-realization process. The approach presented in this article develops shared knowledge about engineering formats and work flows. The format registry captures how these diverse and complex formats interact, enabling long-term interpretability and monitoring of their evolution over time. The engineering work flows formally capture the context required to interpret engineering data over time, as well as to understand vital relationships among datatypes.

Currently, these techniques are actively deployed as part of the CIBER-U National Engineering Dissection Collaboratory. While the techniques exist behind the scenes of the cyber infrastructure used by CIBER-U, they are actively being used by hundreds of engineering design students annually. It is expected that the use of semantic wiki technologies to support this effort will improve the ability of educators to teach engineering design and enable the creation of an extensible library of virtual product dissections.

Beyond CIBER-U, the digital engineering data format registry is being contributed to support the ongoing efforts of several private and government organizations in tracking engineering data elements and their interactions in order to create persistent, 100-year digital engineering archives.

Acknowledgments

This work was supported in part by the Library of Congress National Digital Information Infrastructure Preservation Program (NDIIPP) through National Science Foundation (NSF) DIGARCH Program grant CISE/IIS-0456001. Additional support was provided by the NSF Cyber-Infrastructure Teams (CI-TEAMs) Program grants CISE/SCI-0537370 and CISE/SCI-0537125. Opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or the other supporting government and corporate organizations.

Notes

1. See the 2004 report by the Defense Science Board Task Force on B-52H Re-Engining (www.acq.osd.mil/dsb/reports/ADA428790.pdf).
2. See gicl.cs.drexel.edu/wiki/CIBER-U.
3 See also Pronom at the National Archives of the United Kingdom (www.nationalarchives.gov.uk/pronom).
4. See gicl.cs.drexel.edu/wiki/Briggs_Stratton_Engine.
5. See www.w3.org/TR/owl-ref.
6. See www.w3.org/2006/time.
7. See repository.designengineeringlab.org.
8. See designrepository.org.

References

Abrams, S., and Seaman, D. 2003. Towards a Global Digital Format Registry. Paper presented at the 69th General World Library and Information Congress, IFLA General Conference and Council, August 1–9, Berlin, Germany.

Ashino, T., and Fujita, M. 2006. Definition of a Web Ontology for Design-Oriented Material Selection. *Data Science Journal* 5: 52–63.

Bohm, M. R.; Stone, R. B.; Simpson, T. W.; and Steva, E. D. 2008. Introduction of a Data Schema: To Support a Design Repository. *Computer Aided Design* 40(7): 801–811.

Consultative Committee for Space Data Systems. 2002. Reference Model for an Open Archival Information System (OAIS). ISO Standard 14721:2003 Technical Report, Consultative Committee for Space Data Systems (CCDS), Reston, VA.

Darlington, M. J., and Culley, S. J. 2008. Investigating Ontology Development for Engineering Design Support. *Advanced Engineering Informatics* 22(1): 112–134.

Gil, Y.; Deelman, E.; Ellisman, M.; Fahringer, T.; Fox, G.; Gannon, D.; Goble, C.; Livny, M.; Moreau, L.; and Myers, J. 2007. Examining the Challenges of Scientific Workflows. *Computer* 40(12): 24–32.

Gladney, H. 2006. Principles for Digital Preservation. *Communications of the ACM* 49(2): 111–116.

Grüninger, M., and Fox, M. S. 1995. Methodology for the Design and Evaluation of Ontologies. Paper presented at the IJCAI Workshop on Basic Ontological Issues in Knowledge Sharing, August 20–25, Montreal, Quebec, Canada.

Grüninger, M., and Menzel, C. 2003. The Process Specification Language (PSL) Theory and Applications. *AI Magazine* 24(3): 63–74.

Kopena, J., and Regli, W. 2003. Functional Modeling of Engineering Designs for the Semantic Web. *IEEE Data Engineering Bulletin* 26(4): 55–62.

Krötzsch, M.; Vrandecic, D.; Völkel, M.; Haller, H.; and Studer, R. 2007. Semantic Wikipedia. *Journal of Web Semantics* 5(1): 251–261.

Lemaignan, S.; Siadat, A.; Dantan, J.-Y.; and Semenenko, A. 2006. MASON: A Proposal for an Ontology of Manufacturing Domain. Paper presented at the IEEE Workshop on Distributed Intelligent Systems, June 15–16, Prague, Czech Republic.

Lin, J.; Fox, M.; and Bilgic, T. 1996. A Requirement Ontology for Engineering Design. *Concurrent Engineering* 4(3): 279–291.

Marchese, A.; Ramachandran, R.; Hesketh, R.; Schmalzel, J.; and Newell, H. 2003. The Competitive Assessment Laboratory: Introducing Engineering Design Via Consumer Product Benchmarking. *IEEE Transactions on Education* 46(1): 197–205.

Otto, K., and Wood, K. 2006. *Product Design: Techniques in Reverse Engineering and New Product Development*. Boston: Pearson Custom Publishing.

Ram, S., and Liu, J. 2006. Understanding the Semantics of Data Provenance to Support Active Conceptual Modeling. In *Active Conceptual Modeling of Learning (ACM-L) Workshop,* 17–29. Lecture Notes in Computer Science 4512. Berlin: Springer.

Regli, W. C., and Gaines, D. M. 1997. An Overview of the NIST Repository for Design, Process Planning, and Assembly. *Computer Aided Design* 29(12): 895–905.

Regli, W. C.; Hu, X.; Atwood, M.; and Sun, W. 2001. A Survey of Design Rationale Systems: Approaches, Representation, Capture and Retrieval. *Engineering with Computers* 16(1): 209–235.

Sheppard, S. D. 1992. Mechanical Dissection: An Experience in How Things Work, Paper presented at Engineering Education: Curriculum Innovation and Integration, January 6–10, Santa Barbara, CA.

Simpson, T.; Stone, R. B.; Lewis, K. E.; and Regli, W. C. 2007. Using Cyberinfrastructure to Enhance Product Dissection in the Classroom. Paper presented at the 2007 Industrial Engineering Research Conference, May 19–23, Nashville, TN, USA.

U.S. Library of Congress. 2006. Sustainability of Digital Formats: Planning for Library of Congress Collections. Technical report, Library of Congress, Washington, DC.

Thilmany, J. 2005. Ephemeral Warehouse. *Mechanical Engineering* 127(9): 1–10.

Zhang, S.; Shen, W.; and Ghenniwa, H. 2004. A Review of Internet-Based Product Information Sharing and Visualization. *Computers in Industry* 54(1): 1–15.

William Regli is a professor of computer science at Drexel University with joint appointments in mechanical engineering and electrical and computer engineering. Regli has interdisciplinary research interests, and his contributions span several computer science and engineering fields, including artificial intelligence, network-centered systems, and engineering design and manufacturing. His research has sponsored more than 150 graduate and undergraduate students and produced more than 200 technical publications. Regli is the recipient of many awards, including the NSF Career Award, an NRC Postdoc, NIST Special Service Award, CERDEC Director's Award, and the U.S. Army's 2006 International Collaboration program award. He is a life member of AAAI and Sigma Xi and a senior member of both IEEE and ACM. Contact him at regli@drexel.edu.

Joseph B. Kopena is a researcher in Drexel University's Secure Wireless Agent Testbed Laboratory. His research interests include knowledge representation and wireless networking.

Michael Grauer received his MS in computer science from Drexel Univesity in 2008.

Timothy W. Simpson is a professor of mechanical and industrial engineering at the Pennsylvania State University with affiliate faculty appointments in engineering design and in the College of Information Science and Technology. He is also the director of the Learning Factory. His research interests include engineering optimization, product family and product platform design, and multidimensional data visualization. He has advised more than 50 graduate and undergraduate students and coauthored more than 200 technical papers. He is a recipient of the NSF Career Award, the Penn State President's Award for Excellence in Academic Integration, the SAE Ralph R. Teetor Educational Award, and several outstanding research and teaching awards at Penn State. He is a Fellow in ASME and an Associate Fellow and lifetime member of AIAA. Contact him at tws8@psu.edu.

Robert Stone is a professor in the School of Mechanical, Industrial, and Manufacturing Engineering at Oregon State University. Stone's research interests include design theories and methodologies, specifically product architectures, functional representations, automated conceptual design techniques, and renewable energy systems design. He leads the Design Engineering Lab. Previously, Stone utilized his design background to assist in creating the design-focused interdisciplinary engineering degree programs at the University of Missouri–Rolla. He has authored chapters on product architecture and reverse engineering techniques in product design texts as well as more than 150 technical publications. Prior to initiating his graduate work, Stone worked in the Missions Operation Directorate of NASA–Johnson Space Center as a space shuttle flight controller for the Guidance, Navigation, and Control Section.

Kemper Lewis is a professor of mechanical and aerospace engineering at the University at Buffalo–SUNY and executive director of the NYS Center for Engineering Design and Industrial Innovation. Lewis's research interests include decentralized design, decision-based design, and multiobjective optimization. His research has produced more than 130 refereed publications, and he has been part of more than 6 million dollars in research funding. Lewis was selected for the National Academies Panel on Benchmarking the Research Competitiveness of the U.S. in Mechanical Engineering in 2006 and is the recipient of many awards, including the NSF Career Award, the SAE Ralph R. Teetor Educational Award, and the State University Chancellor's Award for Excellence in Teaching. He is an Associate Fellow in AIAA and a life member of ASME. Contact him at kelewis@buffalo.edu.

Matt Bohm is currently an assistant professor and senior researcher in the School of Mechanical, Industrial, and Manufacturing Engineering at Oregon State University. He joined the school after completing his doctorate in mechanical engineering at the Missouri University of Science and Technology. The thematic focus of Bohm's research is design information archiving and reuse. Recently his work has explored artificial intelligence algorithms to support a new method of automated concept generation. At OSU, Bohm plans to continue research in the area of engineering design, focusing on information data discovery and artificial intelligence–based concept-generation techniques.

David Wilkie is a gradute of Drexel University and former member of the Geometric and Intelligent Computing Laboratory. He is currently a PhD student at the University of North Carolina at Chapel Hill.

Martin Piecyk is an undergraduate student at Drexel University and a member of the Geometric and Intelligent Computing Laboratory.

Jordan Osecki is an undergraduate student at Drexel University and a member of the Geometric and Intelligent Computing Laboratory.