Proceedings of CIE'03
2003 ASME Computers and Information in Engineering Conference
September 2-6, 2003, Chicago, IL

CIE-2003-28

TOWARDS MODELING THE EVOLUTION OF PRODUCT FAMILIES

Fujun Wang, Steven J. Fenves, Rachuri Sudarsan and Ram Sriram
Design and Process Group, Manufacturing Systems Integration Division
National Institute of Standards and Technology, USA
fuwang@cme.nist.gov

ABSTRACT
A strategy successfully used by manufacturing companies is to develop product families so as to offer a variety of products with reduced development costs. This paper introduces our initial research on the representation of the evolution of product families and of the rationale of the changes involved. The information model representing product families is an extension of the NIST Core Product Model and consists of three sub-models: Product Family, Family Evolution, and Evolution Rationale. In addition, an Unified Modeling Language (UML)-based representation and a prototype implementation of the conceptual model are introduced.
Keywords: Product Modeling, Product Variety, Product Family, Product Derivation, Design Evolution, Design Rationale

1. INTRODUCTION
The provision of product variety has become a common practice in today's manufacturing world [6]. Issues arising out of this practice, such as Design for Variety, Variety Management, and Product Family Architecture have been studied by researchers in engineering and business [2,11,12,6,10]. This paper addresses the representation of the product data throughout the evolution of product families and of the underlying rationale for the changes in product families; these two topics have not been adequately addressed in the literature.

1.1 An Illustrative Example
Rather than developing and offering large numbers of unrelated products, manufacturers find it advantageous to deal with families of products, some of long duration. In order to understand the evolution of product families, we first provide a brief review of the evolution of the CFM56 aircraft engine family (all information presented here is available on the CFM Inc. web site: http://www.cfm56.com). CFM56 is a high bypass ratio turbine fan engine, under development since the early 1970s by Snecma Moteurs in France and GE in the U.S.A. The engine's design is flexible in order to meet specific aircraft needs; it consists of four fan sizes, ranging from 60 inches to 72 inches in diameter, and six series, producing thrusts from 18,500 to 34,000 pounds. Figure 1 illustrates the evolution paths of the engine series:

- CFM56-2 is the family baseline, offered since 1972. It was the first high bypass ratio turbine fan engine in the 10-ton thrust class, with a 68.3 inch fan diameter. Its three sub-series supply power for the military aircraft KC-135R, C-135R, E-3, KE-3A, E-6A, and DC-8.
- CFM56-3 was derived from CFM56-2, tailored to meet the needs of short- to medium-range Boeing 737-300/-400/-500 aircraft. The fan diameter is 60 inches, the smallest in the family. Some components were improved, such as the low-noise, low-emission combustor and the elliptical spinner for improved hail/rain ingestion. Even so, the CFM56-3 offers 84 and 60 percent commonality in parts and tooling, respectively, with its predecessor. It has three sub-series, -3-B1, -3B-2, and -3C-1, with thrusts varying from 18,500 to 24,000 pounds.
- CFM56-5A is an advanced derivative engine of the CFM56-2 and CFM56-3. It was developed to power the Airbus A319 and A320 at 22,000 to 26,500 pound thrusts with four sub-series, -5A1, -5A3, -5A4, and -5A5. It has the same fan size as the original –2's. This series is characterized by enhanced efficiency and an improved thermodynamic cycle. Equipped with a three-dimensional aerodynamic fan and a Full Authority Digital Electronic Control (FADEC) that unifies aircraft and engine systems, the CFM56-5A's fuel consumption is 10-11 percent lower than that of its predecessors.

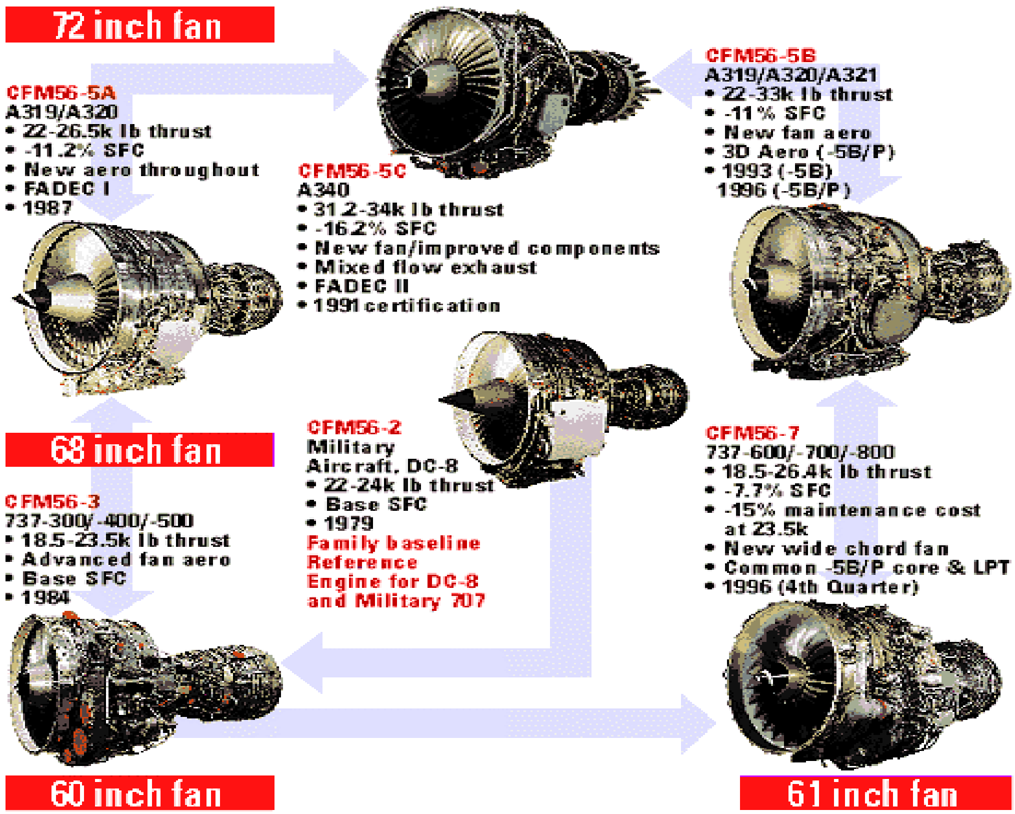

Figure 1 CFM56 Engine Family Evolution Paths (http://www.cfm56.com)

- CFM56-5C, the most powerful engine in the family, is the propulsion system for the long-range Airbus A340-200 and A340-300 aircraft, with a thrust range from 31,200 to 34,000 pounds. It has the largest fan, 72.3 inches in diameter. The nacelle provides significant noise attenuation, reduced fuel burn, and increased climb thrust. In addition, this series adopts a second-generation FADEC.
- CFM56-5B was originally developed to power the Airbus A321. Today, it powers every model in the A320 family with its 9 sub-series. It has the highest fan pressure ratio in the family. Its Double Annular Combustor (DAC) reduces NOx emissions by as much as 45 percent.
- CFM56-7 was developed to provide Next-Generation Boeing 737 aircraft with higher thrust, improved efficiency, and lower maintenance costs than its predecessor, the CFM56-3. It uses the DAC for low emissions capability. This engine uses the common core and low-pressure turbine of the CFM56-5B. One of its featured designs is the wide-chord titanium fan.

The development of each new engine was based on the successful design of its predecessors. This development strategy has led to the following benefits for the company and its customers [18]:

- *Time and Costs Savings.* The product family development approach reduces development time and costs dramatically. For example, it took only 15 months to develop the CFM56-7 from program launch to first engine test, while normally it takes 5 to 8 years to develop a brand new turbine engine [1].
- *Extensive Applications.* The derived engines have extended the family's applications, which now cover short-, medium- and long-range aircraft.
- *Increased Reliability.* Because the new engines were developed based on engines with demonstrated reliability, the CFM56 engines lead the industry in terms of reliability, durability, performance retention, and time-on-wing.
- *Ease of Use and Maintenance.* The commonality across engine models provides airlines with substantial savings in tooling, training, and spare parts inventories.

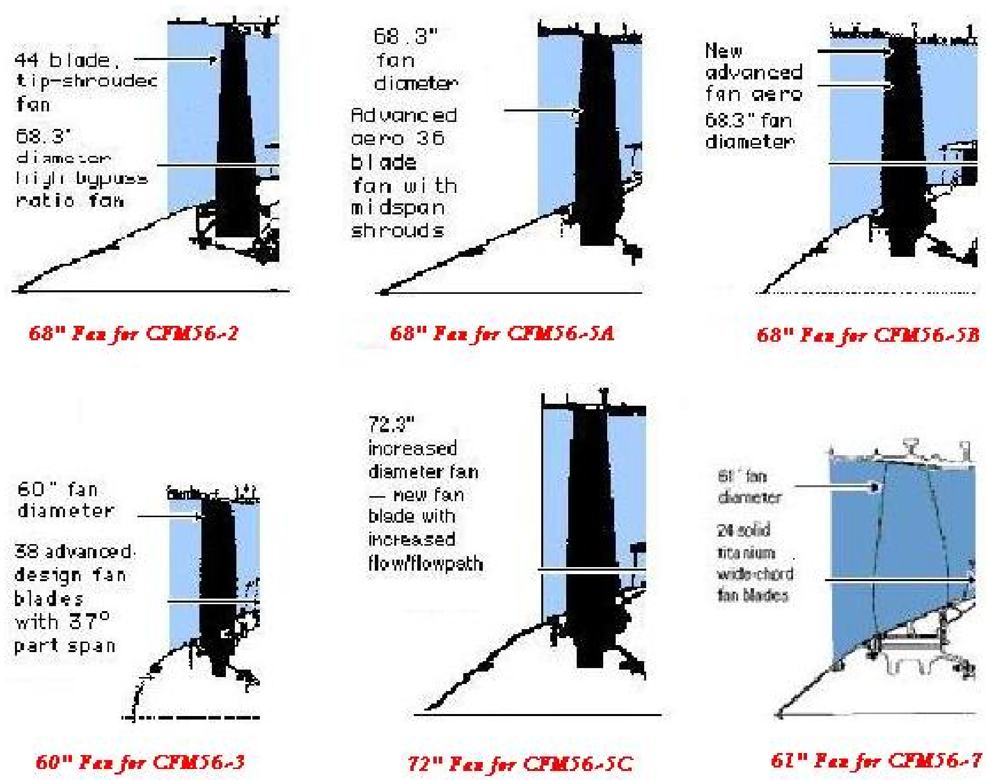

Figure 2 The Fan Family Configured for the CFM56 Engine Family

1.2 Issues In Modeling And Managing Product Families

The design of derived products has to be based on a full understanding of the original base design and the design of their predecessors. Therefore, it is necessary to fully and accurately capture the evolution of a product family.

Commercial Product Lifecycle Management (PLM) systems provide some important, but limited support in this area. Present PLM systems focus on data and process management during the development lifecycle of a single product, and not on entire families' derivation history (http://www.ptc.com, http://www.eds.com). The Change Management component of PLM systems manages data versions and change authorizations during development lifecycle of a product, but again not that of the family [13]. The new Variety Management component of PLM systems provides certain capabilities for representing the product family, but as far as we know, so far no PLM system vendor has provided the capability of representing the full product throughout the product family evolution and the rationale for that evolution.

As noted previously, some issues pertaining to Product Family or Variety have been studied. Ulrich [17] and Erens [2] have defined the product family architecture. Jiao and Tseng [10] proposed fundamentals of product family architecture for mass customization. Martin [11,12] and Fujita [2] investigated approaches to design for variety. Sanderson and Uzumeri [14] presented a way to manage the Sony Walkman product family from the point of view of management science. The above studies of product family architecture and variety are valuable for addressing some of the issues of our research, but the full set of issues of product family representation in the family's evolution have not yet been explored.

In the research community addressing Design Rationale, the rationale for product family evolution has not been investigated [7,15]. The NIST Design Repository manages knowledge-oriented design information such as design rationale of individual products [16]. However, the system does not have provisions for modeling the product families and their evolution.

To overcome the above shortcomings, the emphasis here is on modeling the information reflecting the evolution of product families and of the design rationale driving the changes.

2. THE PRODUCT FAMILY EVOLUTION MODEL

The Product Family Evolution Model (PFEM) is an extension of the NIST Core Product Model (CPM). CPM is a knowledge-oriented design information product model, developed to meet the requirements of a variety of next-generation product development systems [3]. CPM focuses on artifact representation, including the artifact's function, form and behavior, on physical and functional decompositions of the artifact, and on the relationships among these concepts. The PFEM extension consists of three sub-models: family, evolution, and rationale, discussed in the following subsections.

2.1 Product and Component Families

A product is made up of components, which are typically developed and managed by different departments or even outsourced. These components usually have their own family definitions, which may not be identical to the product family's definitions. For example, the fan is a component of the turbine engine. Therefore, the product family and component family need to be modeled separately, and the configuration relationship between a product in the family and its components is an important aspect. In the previously illustrated case, the engine family is designated by CFM56-X, while the fan series are identified by fan size. The four fan sizes, 60", 61", 68" and 72" in diameter, configure for the six engine series as shown in Figure 2. The figure illustrates the variation of form (fan diameter, number of blades, blade profile, etc.) in response to changes in functional requirements (thrust, fuel efficiency, noise attenuation, maintainability, etc., shown in Figure 1).

The definition of a family is not as simple as that of a product list, because there is variation in the products themselves, in the components comprising the product, and in the configuration in which the components are assembled. Before modeling the artifact family, the concepts of "Family Designation" and "Design Representation" needs to be distinguished. Family Designation refers to the family naming space, while Design Representation refers to the design data of an artifact in the family. In this paper, Family Designation is represented by the classes "Family", "Series", and "Version", while Design Representation is represented by the class "PFEM_Artifact". There exists a one-to-one relationship between a family designation and a PFEM_Artifact. Some of the terms defined in this paper may be different from the terms in ISP 10303-41 and 44 [8,9]

The following concepts are needed for modeling the artifact family:

- **Family.** Family is the designation for an entire artifact family, for example, CFM56. An artifact family is collection of a series of artifacts performing the same function, i.e., product family refers to the entire product series, and component family refers to the entire component series.
- **Series**. This term designates the variety in an artifact's function, form, or configuration. A series may have sub-series. Figure 3 shows the CFM56 series hierarchy. These series form a tree structure with Family as the root.
- **Version**. This is a time-dimensioned aspect of the family definition (see Figure 3). The earlier series of CFM56 engines have been revised, year-by-year, and the series' revisions are still in service today. We call both the initial design and its revisions as versions. Versions form a chain structure.
- **PFEM_Artifact**. As noted previously, a PFEM_Artifact holds the artifact's design data. It is a subclass of the "Artifact" defined in CPM. Each PFEM_Artifact has a family designation.
- **PFEM_Abstract_Artifact** and **PFEM_Real_Artifact**. Among the three family designations, i.e., Family, Series, and Version, only the version refers to an artifact with specific design representation of its function, form, behavior, and configuration, while Family and Series refer to abstract artifacts performing a certain function, but without specific form, behavior, or configuration. Therefore, it is necessary to introduce two concepts, PFEM_Abstract_Artifact and PFEM_Real_Artifact, to represent artifacts with different family designations, i.e., Family, Series, and Version. Specifically, Version refers to a PFEM_Real_Artifact, while Family and Series each refer to a PFEM_Abstract_Artifact. In the CFM56 engine family, for example, "fan" is an abstract engine component. It has no specific form, but all fans provide the same function: compressing the airflow into the inner duct and producing additional thrust through the bypass flow.
- **Configuration**. In manufacturing, the term "Configuration" usually refers to the existence of specific components in a product. In this paper, the term has an extended meaning, i.e., it is the mapping relationship between a product version in a product family and each of its configured components' versions in their respective component families. For example, the configuration relationship between CFM56 engine and fan defines which fan version is configured into which CFM56 engine version.
- **Derivation Relationship**. Derivation relationships exist between elements of the hierarchically structured series as well as between two consecutive versions. A series might derive from one or more series. For example, the CFM56-7 engine derives from both the CFM56-3 and CFM56-5B engines, as shown in Figure 1, but this relationship is not reflected in the tree hierarchy of Figure 3. The bi-directional arrows in Figure 1 indicate that the series are derived from each other. For example, the early sub-series of CFM56-7 are derived from early sub-series of CFM56-5B, while some later sub-series of CFM56-5B are derived from sub-series of CFM56-7. Thus, a sub-series may be derived from sub-series of different parent series.

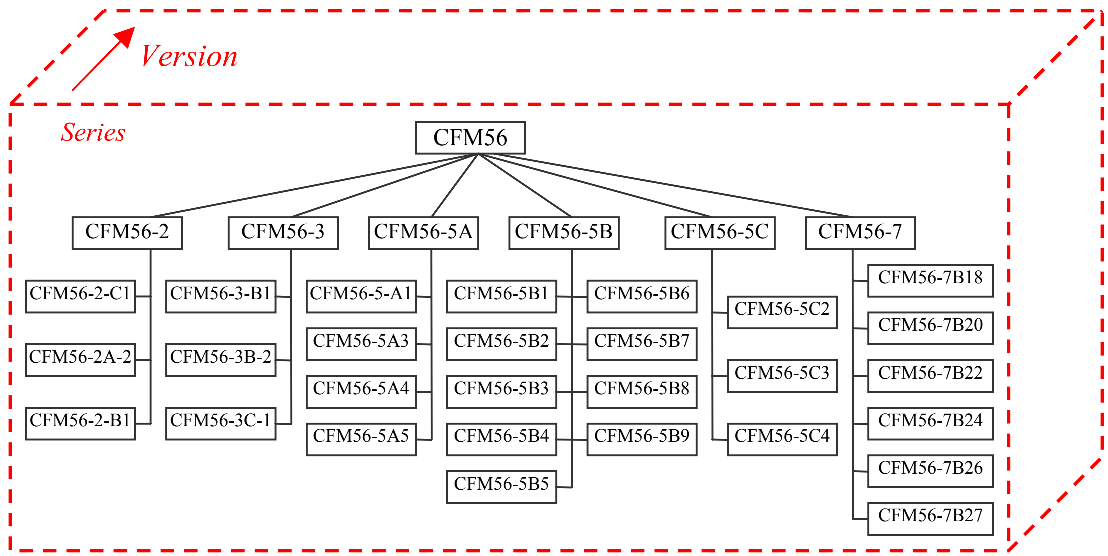

Figure 3 CFM56 Engine Family

2.2 Family Evolution

Family Evolution is the second sub-model of PFEM. It consists of two aspects:

- **Family Derivation** refers to the set of precedence relationships between derivative series and versions in the evolution of the product line. The derivation relationships could be generated by analyzing the linkages among series and versions discussed above, but for the sake of efficiency of retrieval as well as one-to-one parallelism with the following aspect, this redundant representation is used. Informally, family derivation describes, "what derived from what."
- **Design Evolution** contains the design data that have changed between particular series or versions and their predecessor(s). Design data as used here refer to the function, form, and behavior of the artifact defined in the CPM. Informally, design evolution describes "how the derivative designs changed."

Since a product is made up of components, the design evolution of a product equals the combination of component evolution and the configuration evolution.

2.3 Evolution Rationale

While Family Evolution captures what has changed, Evolution Rationale captures the reasons for the changes. Similar to the family evolution classification discussed in the previous subsection, the evolution rationale includes two aspects:

- **Family Derivation Rationale** captures the driving factors for the changes in the product line or, informally, "why a new series or version was introduced" (e. g., because of new requirements, new regulations, or the availability of new technology). Martin [12] calls the driving factors External Drivers of Change. He classifies these drivers as Customer Requirements, Unit Cost, and Regulations. Cost reduction is a perennial aim of a manufacturer, and is thus not a sufficiently precise designation of a driving factor. A new product series or version with lower cost may be possible due to a new design capability, a new manufacturing technique or other technology advance. Consequently, the family derivation factors considered in the model are **Requirement**, **Technology** and **Regulation**.
- **Design Evolution Rationale** records the reason for design changes between a series or version and its predecessor.

3.0 UML-BASED PRODUCT FAMILY EVOLUTION MODEL

A conceptual PFEM is under development by our group at NIST. This section illustrates the model represented in the Unified Modeling Language (UML). UML is a standard object modeling language [4] which has been widely used in object-oriented system design and analysis. In what follows, PFEM class names are shown in **bold**.

Figure 4 shows the main diagram of the PFEM. The package **NISTCoreProductModel+** reuses the classes defined in CPM and contains extension classes defined in other projects. The sub-models of PFEM, namely, Family, Evolution and Rationale discussed above, are defined in the packages **Family**, **Evolution**, and **Rationale**, respectively. The dependency relationships (represented by dashed arrows) among these packages show that there exist certain relationships (e. g., association or generalization) among classes in different packages. The following diagrams and paragraphs introduce more detailed descriptions of these packages.

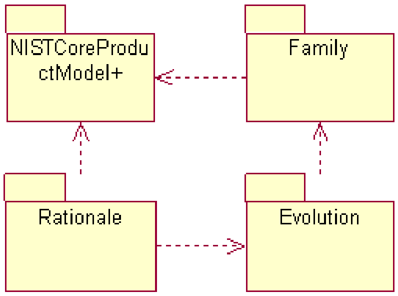

Figure 4 Main Diagram of Product Family Evolution Model

Figure 5 shows the package **NISTCoreProductModel+,** an extension of the NIST Core Product Model (CPM). In the CPM presented in NISTIR 6736 [3], a product is represented by a hierarchy of entities of the class **Artifact,** which is an aggregation of **Function**, **Form** and **Behavior**. **Function** represents what the artifact is supposed to do; **Form** represents the proposed design solution for the design problem specified by the function, and **Behavior** represents how the artifact implements its function. **Form** itself is the aggregation of **Geometry**, the spatial description of the artifact, and **Material**, the internal composition of the artifact. Lower-level Artifact entities may be labeled as **Features**. All the above entities have their own containment ("part-of") hierarchies that are independent. A number of relationships is also defined; the most important relationships are the **Requirement** relationship between the Artifact's **Specification** and some specified aspect of Function or Form, and the **Constraint** relationship on one or more such aspects. The extension of CPM in the package consists of the **DesignRationale** and **AssemblyRelationship** classes.

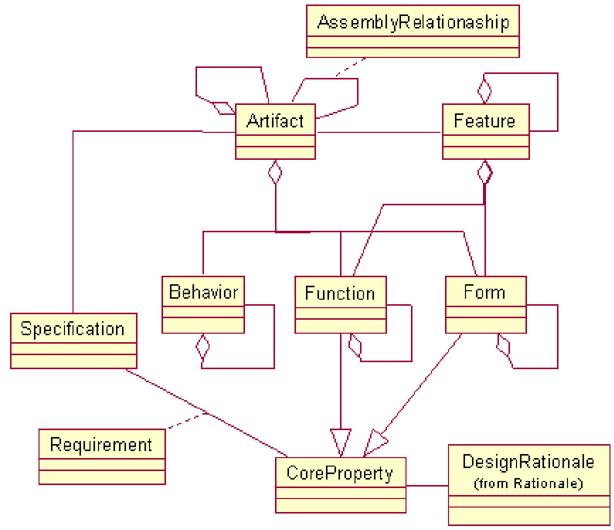

Figure 5 The NISTCoreProductModel+ Package

Figure 6 shows the package **Family**. The class **Family** is an aggregation of **Series**, and **Series** is decomposable. **Version** is associated to the **Series**. A series has a chain of versions. The class **PFEM_Artifact** is a subclass of the class **Artifact** defined in the **NISTCoreProductModel+** package. As a result, **PFEM_Artifact** inherits all attributes of **Artifact**. **PFEM_Abstract_Artifact** and **PFEM_Real_Artifact** are subclasses of **PFEM_Artifact**. There is a one-to-one association between **PFEM_Real_Artifact**, representing the design information of an artifact, and **Version**, representing its designation. In an identical fashion, there is a one-to-one association between **PFEM_Abstract_Artifact**, representing the function of an abstract artifact, and its designation, **Family** or **Series**.

Because product and component families are modeled separately, **ProductFamily** and **ComponentFamily** (not shown in the diagram) are subclasses of **Family**. Correspondingly, product series and versions are defined by **ProductSeries** (not shown in the diagram) and **ProductVersion**; and component series and versions by **ComponentSeries** (not shown in the diagram) and **ComponentVersion**. **Configuration** is the association class between **ProductVersion** and **ComponentVersion**, which defines the configuration between product and component versions.

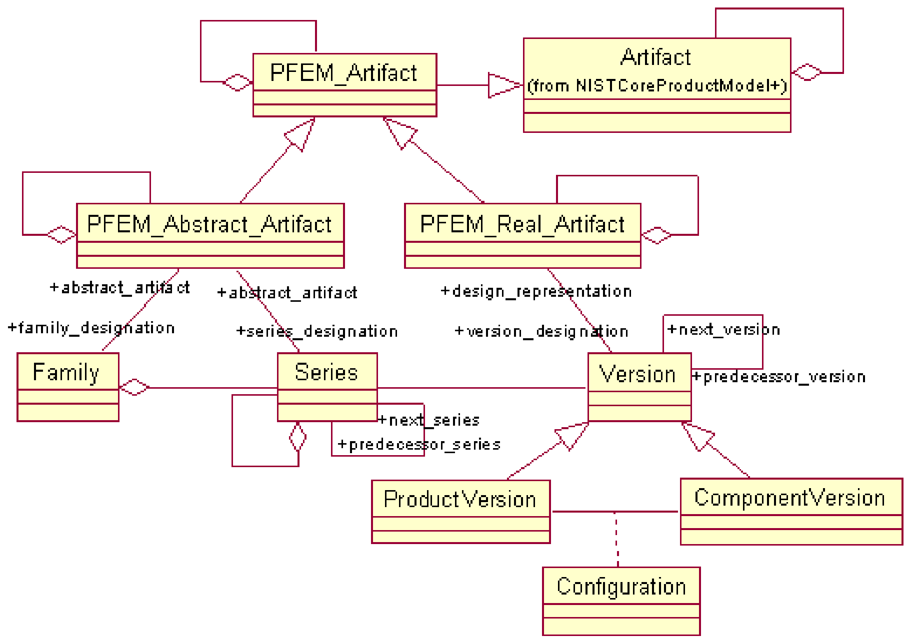

Figure 6 The Family Package

Figure 7 shows the package **Evolution**. The Family Derivation and Design Evolution discussed previously are represented by the classes **FamilyDerivation** and **DesignEvolution**, respectively. As noted, **Evolution** records the derivation and design changes of a family member from its predecessor(s), and is the aggregation of **FamilyDerivation** and **DesignEvolution**. Since the **Family** consists of **Series** and **Versions**, **Evolution** is specialized into the classes **SeriesEvolution** and **VersionEvolution,** respectively. **SeriesEvolution** is the association class between a series and its predecessor series, and **VersionEvolution** is the association class between a version and its predecessor version.

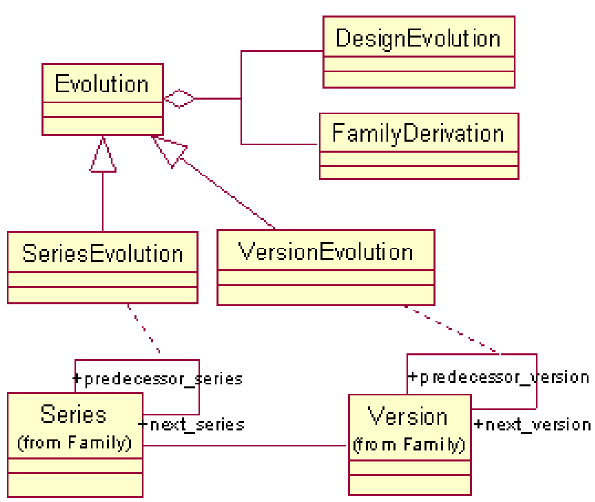

Figure 7 The Evolution Package

The class **Rationale** is defined in the package **Rationale,** which is shown in Figure 8. The classes **Designer**, **DesignDate**, and **ExcludedAlternative** are generic attributes of the class **Rationale**. The classes **DesignRationale** and **EvolutionRationale** are subclasses of **Rationale**. The class **DesignJustification** defines the design justification that could be a requirement, a function, a constraint, a case, a rule, a catalog, a principle, an authority, a trade-off, or a pareto optimal surface.

The classes **DesignEvolutionRationale** and **FamilyDerivationRationale** are subclasses of **EvolutionRationale**, representing the design evolution rationale and family derivation rationale, respectively. The design evolution driving factors are the justifications of changes in the design. The derivation driving factors are described by the class **DevelopmentSpecificationEvolution** which represents the evolution of development specifications. The classes **Requirement**, **Regulation**, and **Technology** are subclasses of **DevelopmentSpecification**. **Requirement** represents the customer requirements; **Regulation** refers to general design regulations or standards governing the design; and **Technology** defines the approaches or tools to solutions. As noted previously, the evolution of these factors is the root reason driving the family derivation.

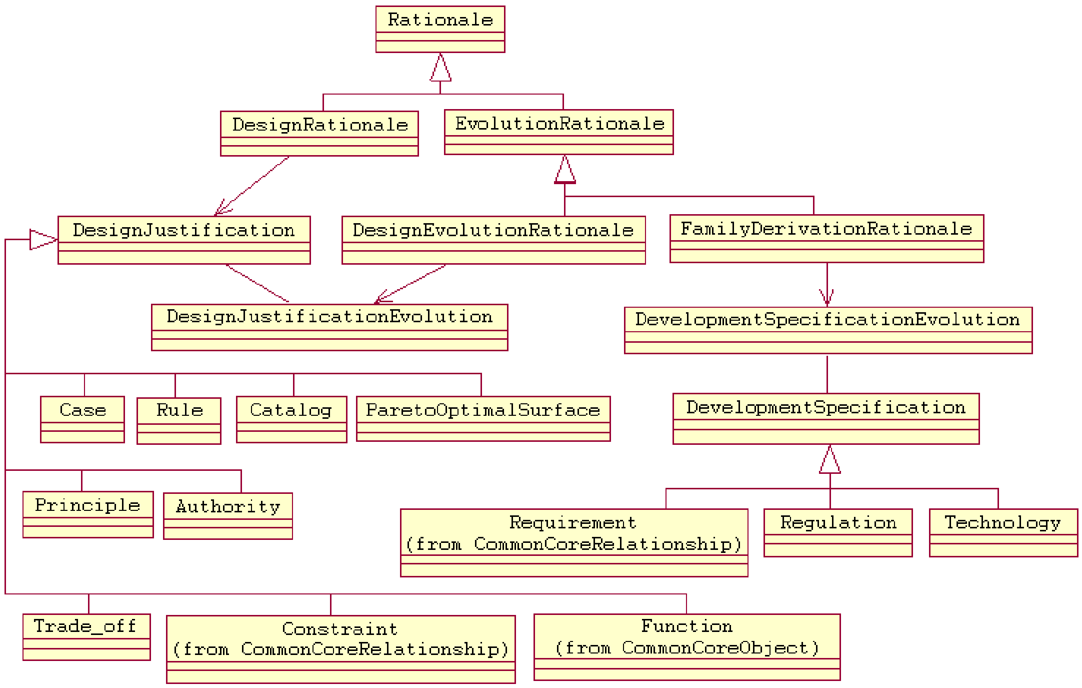

Figure 8 The Rationale Package

4. SYSTEM PROTOTYPE

A system prototype called **EvolutionManager,** has been developed based on the model described above. It is a web-based multi-tier system, written in Java, JSP, JavaScript, and HTML. The prototype's current capabilities include capturing and browsing of the evolution of a product and component family, and of the component configuration in a product family.

Figure 9 shows the prototype's interface. The particular snapshot shows the fan blade configuration in the CFM56 engine revision of CFM56_2A_1980. The left frame is the general component hierarchy of the CFM56 engine, i.e., a tree of abstract components. The central frame lists the available engine series. The right upper frame shows the revision chain of an engine series. The right bottom frame illustrates the configuration information on a component (fan blade) in the product revision (CFM56_2A_1_1980). This configuration interface shows that the fan blade revision of FanBlade_Solid_1979 was configured into the engine revision of CFM56_2A_1_1980. The HTML page of configuration rationale is linked to this page. The configured fan blade quantity, design rationale, and its evolution and evolution rationale compared to its predecessor are shown.

5. CONCLUSIONS

The management of the proliferation of new products by means of product families has become common practice in today's manufacturing world. In order to support the evolutionary design of product families, it is necessary to record and manage the information representing family evolution. A standardized formal representation of the evolution of product families is a current research activity at NIST. This paper introduces initial research results on this issue.

The salient points of the paper are:

1) The Product Family Evolution Model (PFEM) is an extension of the NIST Core Product Model.
2) The PFEM consists of three sub-models: Family, Evolution, and Rationale.
3) Product and component families are modeled and managed separately.
4) Families are modeled by family designation and design representation.
5) The modeling of family evolution includes family derivation and design evolution.
6) The factors driving the evolution are the key in modeling evolution rationale. For the present, we have identified the evolution of requirements, regulation, and technology as the root factors driving product family evolution.

The paper presents the UML-based conceptual representation of PFEM. In order to verify this model, a system prototype is under development.

Future work of the project will focus on model improvement, the development of services supporting design evolution, and calibration of the model with industrial counterparts.

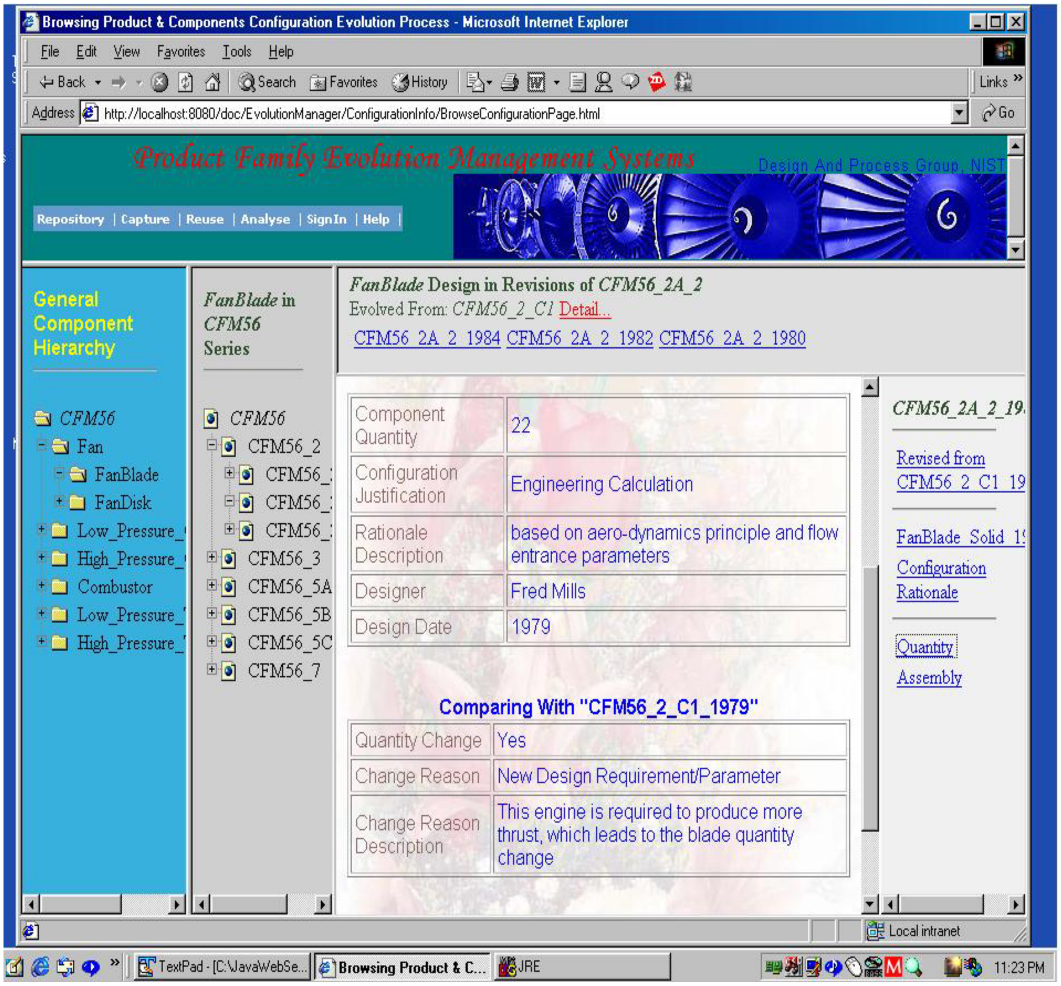

Figure 9 The System Prototype Interface

ACKNOWLEDGEMENT

This project is funded by NIST's Systems Integration for Manufacturing Applications (SIMA) Program. SIMA supports NIST projects applying information technologies and standards-based approaches to manufacturing software integration problems.

DISCLAMER

Commercial equipment and software, if any, are identified only in order to adequately specify certain procedures. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

REFERENCES

1. Chen, G., Ma, M., Xiao, L., and Huang, Z. (1995), "Gas Turbine Design", Beijing University of Aerospace & Astronomics Press, 2nd Edition.
2. Erens, F. and Verhulst, K. (1997), "Architectures for Product Families", Computers In Industry, 33, pp. 165-178.
3. Fenves, S. (2001), "A Core Product Model for Representing Design Information", NISTIR 6736, NIST, Gaithersburg, MD.
4. Fowler, M. (1999), "UML Distilled: A Brief Guide to the Standard Object Modeling Language", 2nd edition, Addison-Wesley Publishing Company.
5. Fujita, K. (2002), "Product Variety Optimization Under Modular Architecture", Computer Aided Design, 34, pp. 953-965.
6. Ho, T. and Tang, C. (1998), "Product Variety Management", Kluwer Academic Publishers, Boston, MA.
7. Hu, X., Pang, J., Pang, Y., Atwood, M., Sun, W., and Regli, W. (2000), "A Survey on Design Rationale: Representation, Capture and Retrieval", DETC2000/DFM-14008, Proceedings of the 2000

ASME Design Engineering Technical Conference, Baltimore, Maryland.
8. ISO, (1994, a), Industrial Automation Systems and Integration – Product Data Representation and Exchange – Part 41: Integrated Generic Resources: Fundamentals of Product Description and Support, ISO 10303-41:1994.
9. ISO, (1994, b), "Industrial Automation systems and integration – Product Data Representation and Exchange – Part 44: Integrated Generic Resources: Product Structure Configuration", ISO 10303-44:1994.
10. Jiao, J. and Tseng, M. (2000), "Fundamentals of Product Family Architecture", Integrated Manufacturing System, 11, No. 7, pp. 469-483.
11. Martin, M. and Ishii, K. (1997), "Design for Variety: Development of Complexity Indices and Design Charts", Proceedings of the 1997 ASME Design Engineering Technical Conferences, Sacramento, CA.
12. Martin, M. (2000), "Design for Verity: A Methodology for Developing Product Platform Architectures", National Manufacturing Week, Stanford University.
13. OMG (2001), "PDM Enablers V2.0", OMG Document mfg/2001-06-02.
14. Sanderson, S. and Uzumeri, M. (1995), "Managing Product Families: The Case of The Sony Walkman", Research Policy, 24, pp. 761-82.
15. Sriram, R. (2002), "Distributed & Integrated Collaborative Engineering Design", Sarven Publishers, Glenwood, MD.
16. Szykman, S., Racz, J., Bochenek, C., and Sriram, R. (2000), "A Web-based System for Design Artifact Modeling", Design Studies, 21, No.2, pp. 145-165.
17. Ulrich, K. (1995), "The Role of Product Architecture In The Manufacturing Firm", Research Policy, 24, pp. 419-440.
18. Wang, F. (1994), "The Design Evolution of CFM56 Engine Families", Graduation Thesis, Beijing University of Aerospace & Astronautics.