CAD Reusability and the Role of Modeling Information in the MBE Context

Jorge D. Camba
jdorribo@uh.edu
UNIVERSITY of HOUSTON

Manuel Contero
mcontero@upv.es
UNIVERSIDAD POLITECNICA DE VALENCIA

Pedro Company
pcompany@uji.es
UNIVERSITAT JAUME I

Agenda

01
Context and Background
- MBE
- Digital Thread

02
Key Terms
- Product Model
- Communication
- Reuse

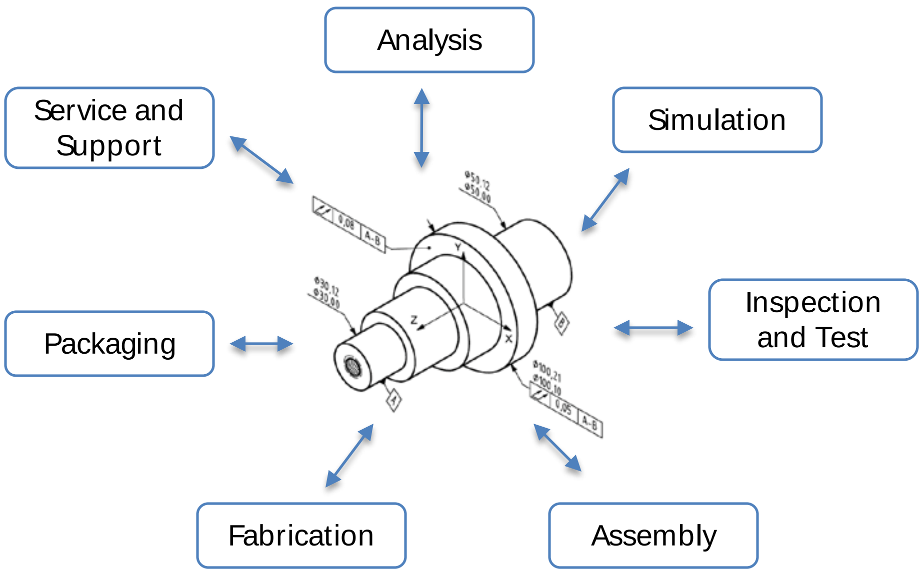
03
Software Analogy
- Source Code
- Documentation
- Software Reuse

04
Vision
- Quality Modeling
- Quality Testing
- Model Enrichment

05
Contributions
- Demo

Model-Based Enterprise

- A Model-based Enterprise (MBE) is an organization that applies modeling and simulation technologies to integrate and manage all of its technical and business processes related to production, support, and product retirement[1].
- It applies product and process models at every step of the product life-cycle
- The core MBE tenet is that data is created once and directly reused by all data consumers[1].



[1]Frechette, S.P. (2011). *Model Based Enterprise for Manufacturing*. 44th CIRP Int. Conf. on Manufacturing Systems. Madison, WI.

Digital Thread

- The Digital Thread for Smart Manufacturing project enables the repurposing, reuse, and traceability of information throughout the product lifecycle…Research focuses on standards and implementation needs to exchange information between each phases of the lifecycle - particularly between engineering, manufacturing, and quality functions[1].

[1] NIST, Systems Integration Division. *Enabling the Digital Thread for Smart Manufacturing* (https://go.usa.gov/6nPh)

“Over-the-wall” Design

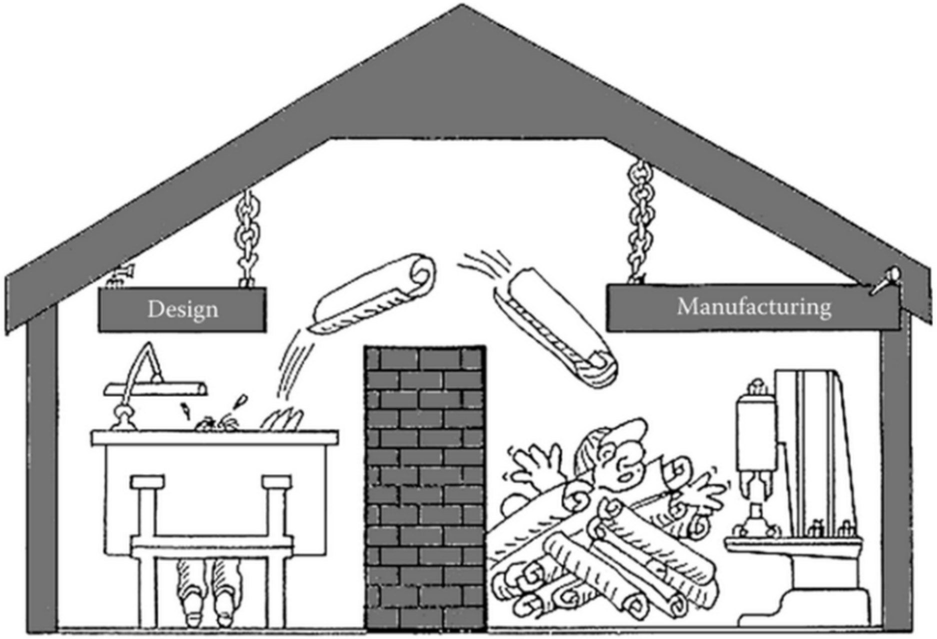
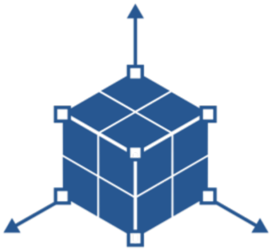


FIGURE 1.6
“Over-the-wall” design, historically the way of doing business.

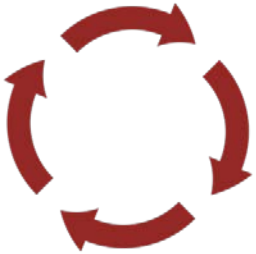
Source: Boothroyd, G., Dewhurst, P., & Knight, W. A. (2010). *Product design for manufacture and assembly.* CRC press.

Key Terms



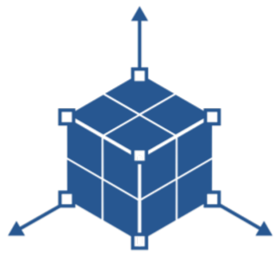
Product Model

Communication



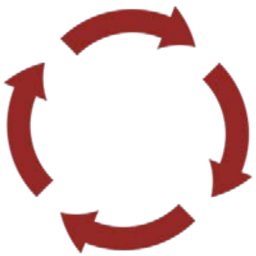
Reuse

Key Terms



Product Model

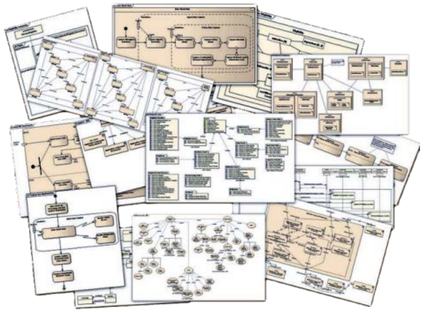
Communication



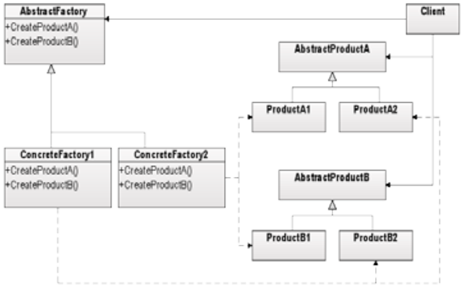
Reuse

An Analogy: Software Engineering

Source Code



Documentation



Software Reuse

An Analogy: Software Engineering

Software Engineering Sub-Disciplines

- Software requirements
- Software design
- Software construction
- Software testing
- Software maintenance
- Software configuration management
- Software engineering management
- Software development process
- Software engineering models and methods
- Software quality
- Software engineering professional
- Software engineering economics
- Computing foundations
- Mathematical foundations
- Engineering foundations

An Analogy: Software Engineering

Software Quality

- The way a program is written can have important consequences.

- Software quality refers to:
  - Functional quality
  - Structural quality

- International standard for the evaluation of software quality: ISO/IEC 25010:2011
  - Functionality, Reliability, Usability , Efficiency, Maintainability, and Portability

- Examples of Strategies:
  - Coding conventions
  - Code refactoring
  - Programming Paradigms (Object-Oriented, Structured, etc.).

Product Model

Product Model

- Just like the quality of a software system depends on the correctness and efficiency of its code, the quality of products depends on the quality of their design processes, which then depend on the quality of their data.

- Product Data Quality (PDQ) is a measurement of the accuracy and appropriateness of all data involved in the design and manufacturing of a product combined with the timeliness with which they are provided to the stakeholders who may need them[1,2].

- Current Model Quality Tools (MQT) are primarily aimed at preventing "easily" solvable low semantic level mistakes and incoherencies[3].

[1] Yang, J., et al. (2006) *Product data quality assurance for e-manufacturing in the automotive industry.* Int. Journal of Computer Integrated Manufacturing; 19(2): 136-147.

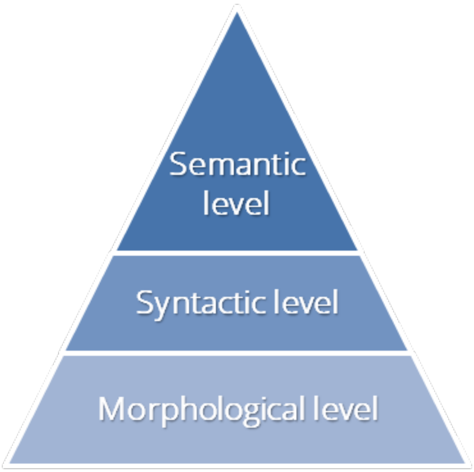
[2] Strategic Automotive product data Standards Industry Group, SASIG, Product Data Quality Workgroup (PDQ) (2005). *SASIG– Product Data Quality Guidelines for the Global Automotive Industry, Guideline version 2.1* (STEP Part 59. ISO/PAS 26183:2006.)

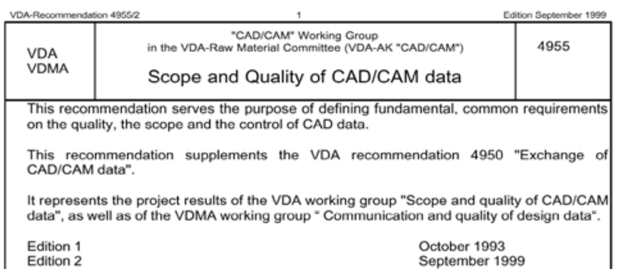
[3] González-Lluch, C., Company, P., Contero, M., Camba, J. D., & Plumed, R. (2017). *A survey on 3D CAD model quality assurance and testing tools*. Computer-Aided Design, 83, 64-79.

Product Model

Product Data Quality Model[1]

- **Morphological quality**: related to the geometrical and topological correctness of the CAD model
- **Syntactic quality**: evaluates the use of proper modeling conventions.
- **Semantic/pragmatic quality**: takes into account the model's capability for reuse and modification





VDA-Recommendation 4955/2 1 Edition September 1999

VDA VDMA | "CAD/CAM" Working Group in the VDA-Raw Material Committee (VDA-AK "CAD/CAM") Scope and Quality of CAD/CAM data | 4955

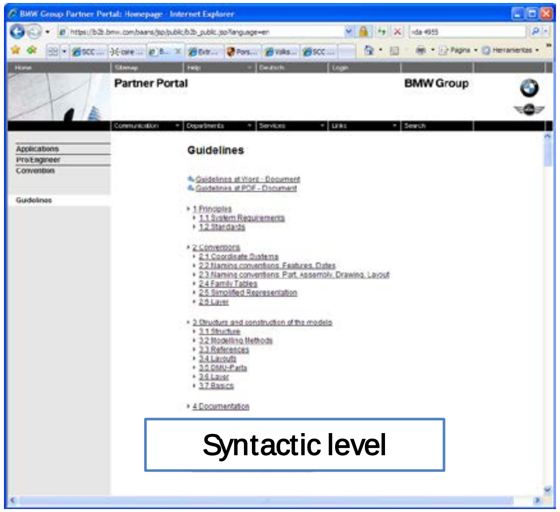
This recommendation serves the purpose of defining fundamental, common requirements on the quality, the scope and the control of CAD data.

This recommendation supplements the VDA recommendation 4950 "Exchange of CAD/CAM data".

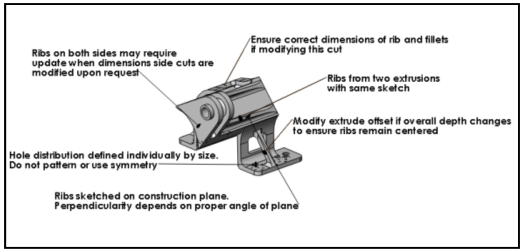
It represents the project results of the VDA working group "Scope and quality of CAD/CAM data", as well as of the VDMA working group " Communication and quality of design data".

Edition 1 October 1993
Edition 2 September 1999

Morphological level



Syntactic level



Semantic level

[1] Contero, M., Company, P., Vila, C., & Aleixos, N. (2002). *Product data quality and collaborative engineering*. IEEE Computer Graphics and Applications, 22(3), 32-42.

Product Model

An Analogy: Software Engineering

Communication

- In software engineering, developers use source code comments to support program maintenance and future modifications.

Source code example from Wikipedia

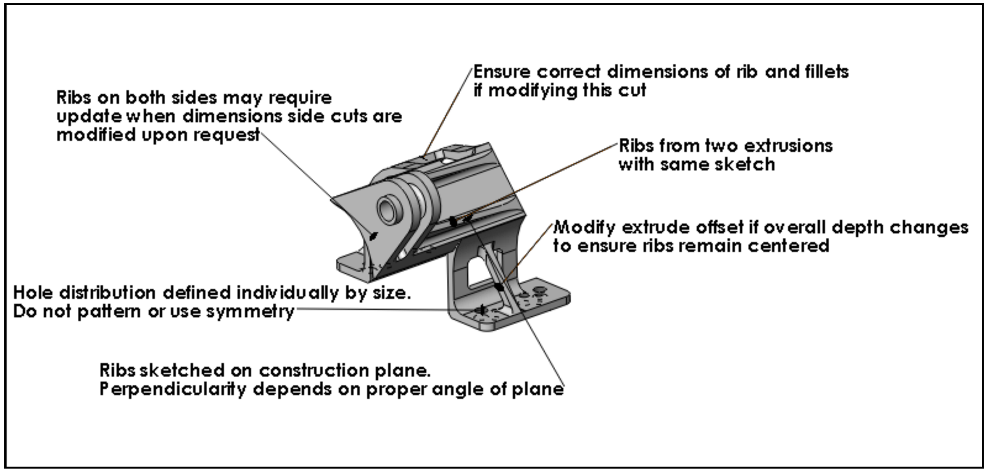
```
/* loop backwards through all elements returned by the
server (they should be processed chronologically)*/

for (i = (numElementsReturned - 1); i >= 0; i--)
   {
    /* process each element's data */
    updatePattern(i, returnedElements[i]);
   }
```

- Tools such as **Doxygen, Javadoc, Apple's HeaderDoc,** or **Microsoft's Sandcastle** can automatically generate project documentation from source code comments.

Communication

- Product and Manufacturing Information (PMI)
  - Geometric Dimensioning and Tolerancing (GD&T)
  - 3D annotations (text)
  - Surface finish
  - Material specifications
- Standards for 3D part definition:
  - ASME Y14.41
  - ISO 16792



Communication

An Analogy: Software Engineering

Software Reusability

- "Code reuse is the Holy Grail of Software Engineering."

  - Douglas Crockford (Senior JavaScript Architect at PayPal)

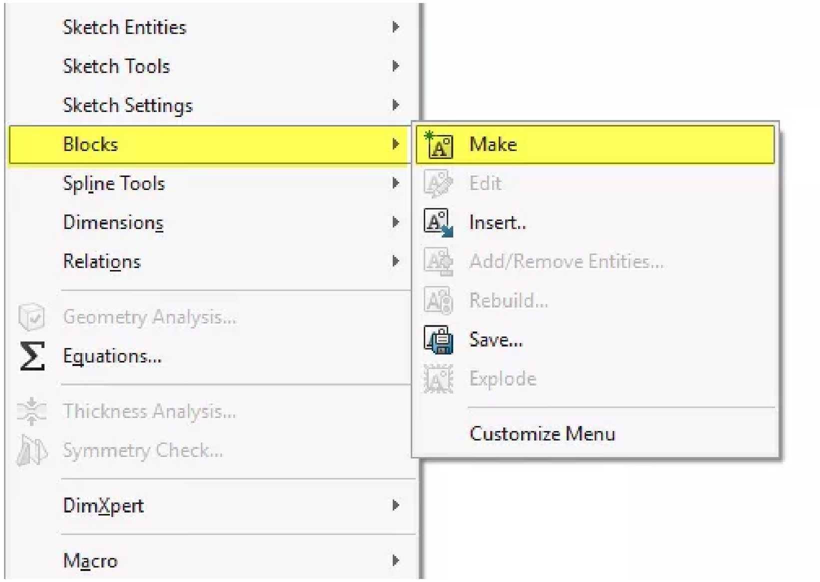
- Reusability is the use of existing assets in some form within the software product development process, including code, software knowledge, test suites, designs and documentation.

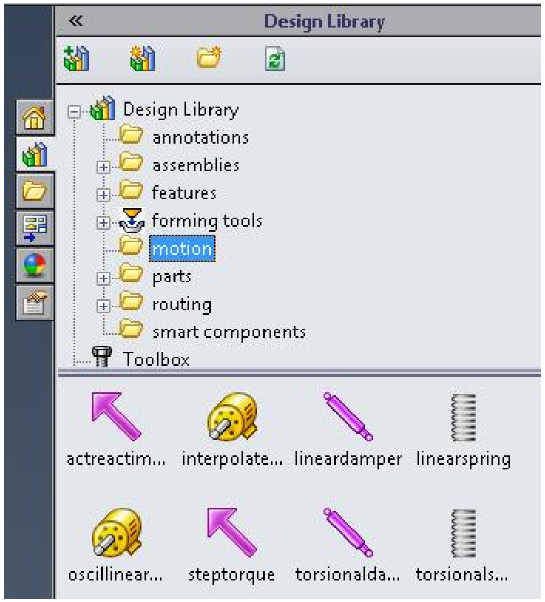
- Examples of Strategies:
  - Subroutines
  - Software Libraries and Components
  - Design Patterns

Reuse

Design Reuse

- Standard Parts and Libraries



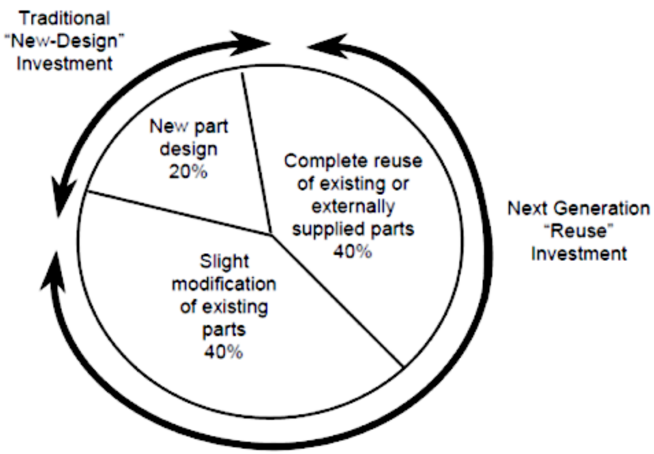


Reuse

Design Reuse

- All engineering organizations report that they reuse designs, but the top performing ones deploy techniques and technologies to CAPITALIZE on design reuse.
- Design Reuse → Model Reuse

Components used in a product.



Rezayat, M. (2000). *Knowledge-based product development using XML and KCs*. Computer-aided design, 32(5), 299-309.

Current Use and Future Plans for Reuse Initiatives

Reuse Initiatives	Currently Use	Plan to Use
Reuse of product designs	80%	20%
Reuse of simulation / analysis content	35%	24%
Reuse of NC programming content	29%	23%

Aberdeen Group, *The Design Reuse Benchmark Report* (2007)

Reuse

Design Reuse

- There are important obstacles to implement CAD model reuse

Challenges	
Model modification required expert CAD knowledge	57%
Models are inflexible and fail after changes	48%
Only original designer can change models successfully	40%

Aberdeen Group. *The Design Reuse Benchmark Report (2007)*

Reuse

Design Reuse

- There are important obstacles to implement CAD model reuse

Challenges	
Model modification required expert CAD knowledge	57%
Models are inflexible and fail after changes	48%
Only original designer can change models successfully	40%

Responses	
Train users to increase CAD skills	71%
Design for wide range of modifications	64%
Detail design information in model (PMI)	36%

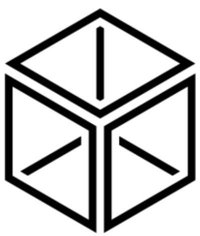
Aberdeen Group. *The Design Reuse Benchmark Report (2007)*

Reuse

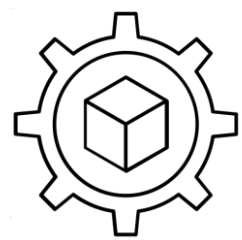
Vision

- Improving digital product representation and data quality and its connection to design knowledge at the early stages of the product life-cycle can reduce the complexity of the "digital thread" and lead to more efficient product development processes.

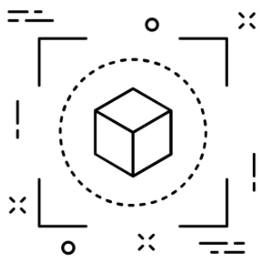
- The execution of this vision is based on three primary ideas:



High Quality Modeling

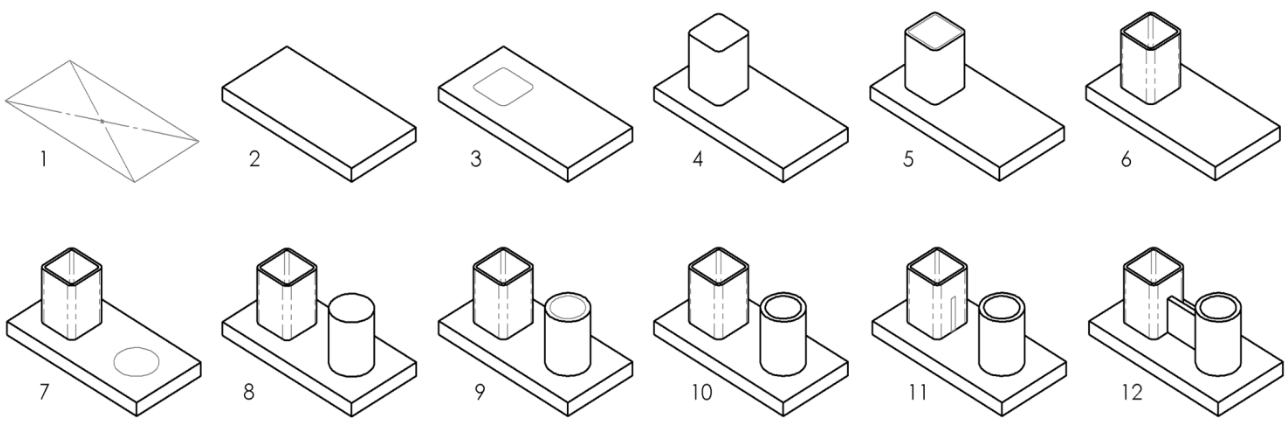


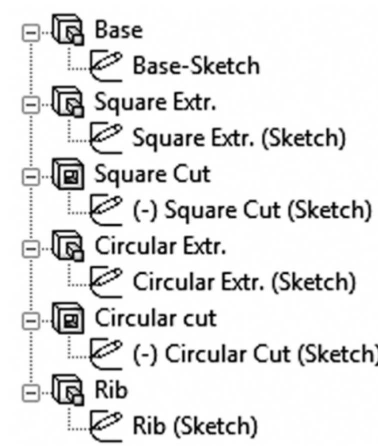
Model Quality Testing



Model Enrichment

- History-based parametric CAD modeling

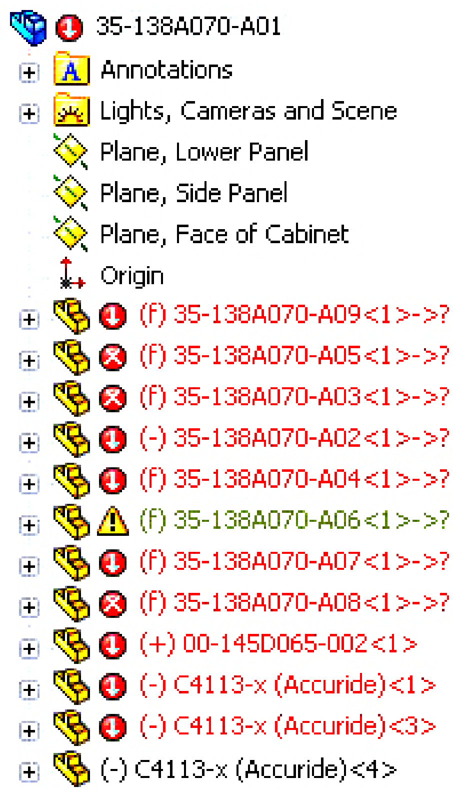




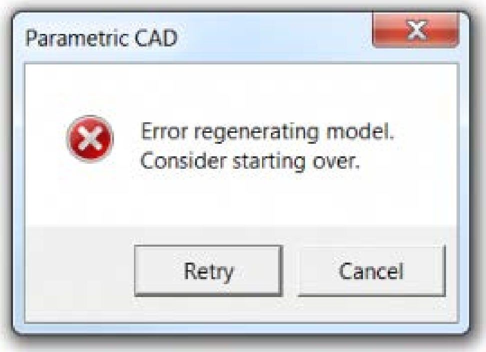
Design Tree
(SolidWorks)

1. High Quality Modeling

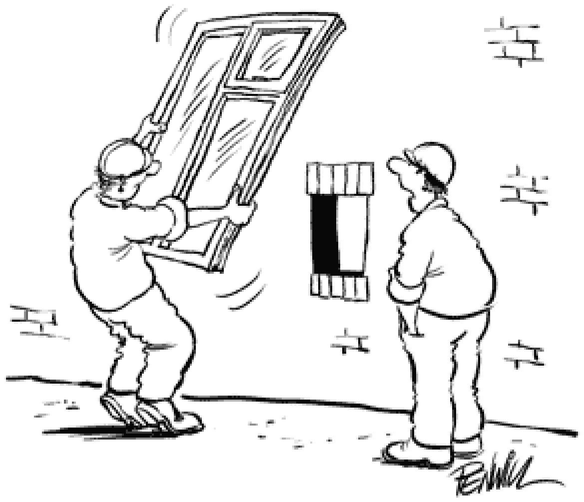
- Rebuild Errors



1. High Quality Modeling



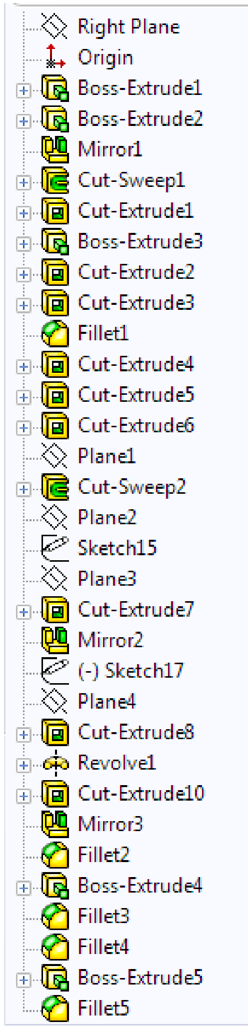
Source: Evan Yares, *The failed promise of parametric CAD* (http://www.3dcadworld.com/the-failed-promise-of-parametric-cad/)



© Roger Penwill

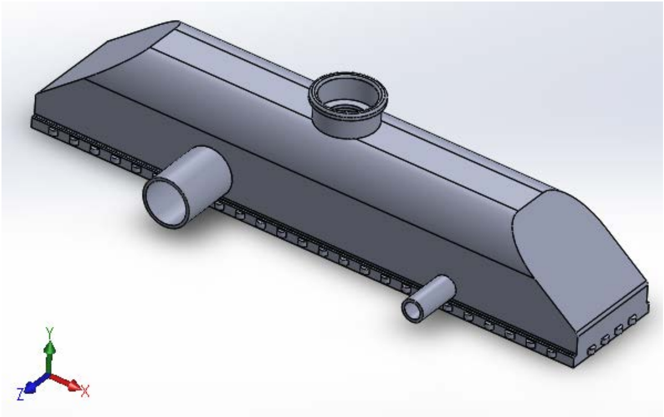
Design Intent

- In parametric CAD, design intent is expressed implicitly in the model's design tree.
- The size and underlying complexity of the design tree can grow rapidly, even for relatively simple models.
- Modeling decisions are not explained.



1. High Quality Modeling

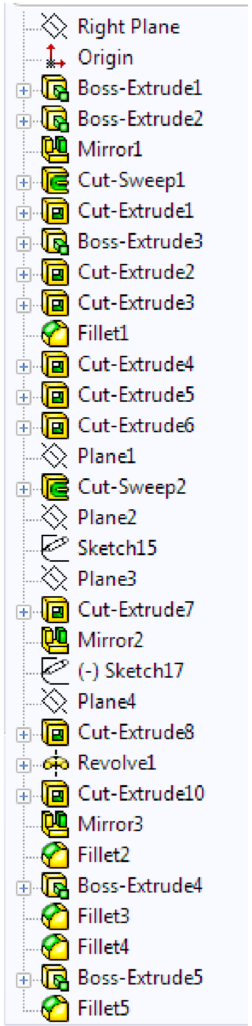
- History-based parametric CAD modeling (behind the scenes)



Total Features: 48
Total Dependencies: 97

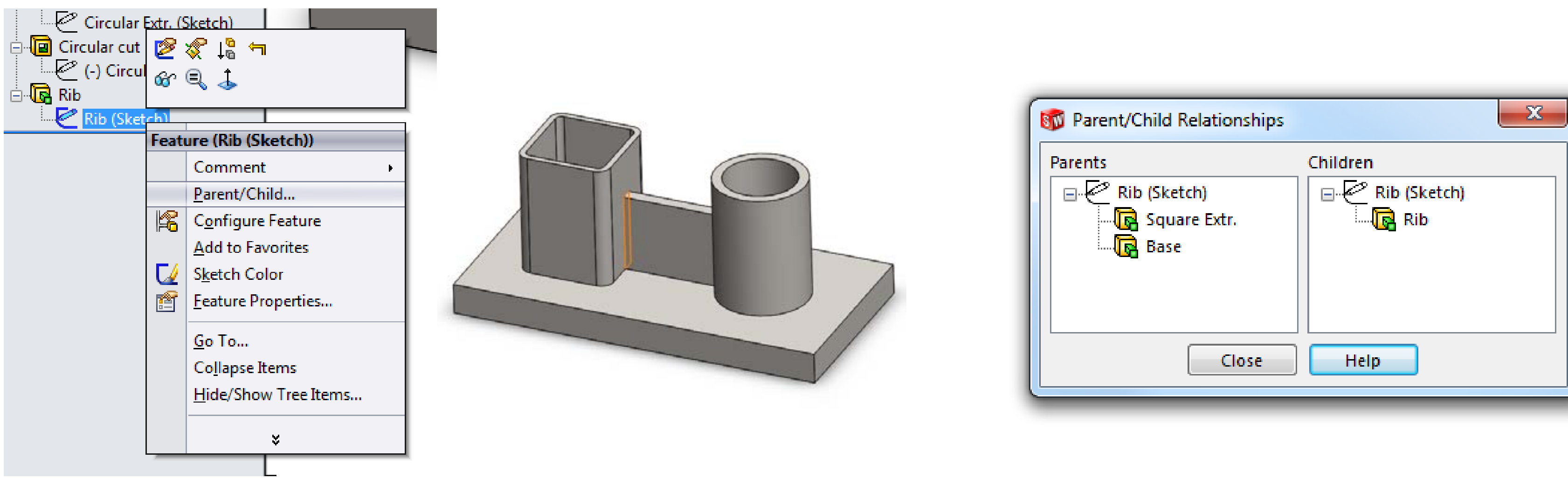
1. High Quality Modeling

- Parent/child relationships can be problematic.
- The interdependencies of the model should be defined according to the design intent.
- But the design tree is only a sequential list of features…



Strategies

- Improving the understanding of parent-child relationships

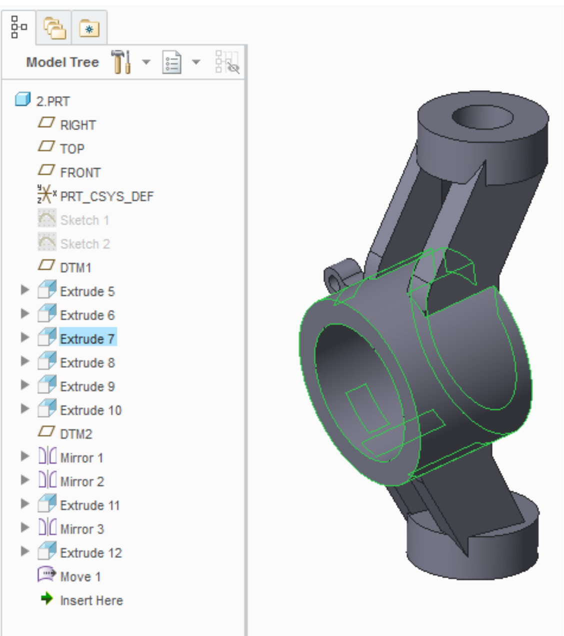


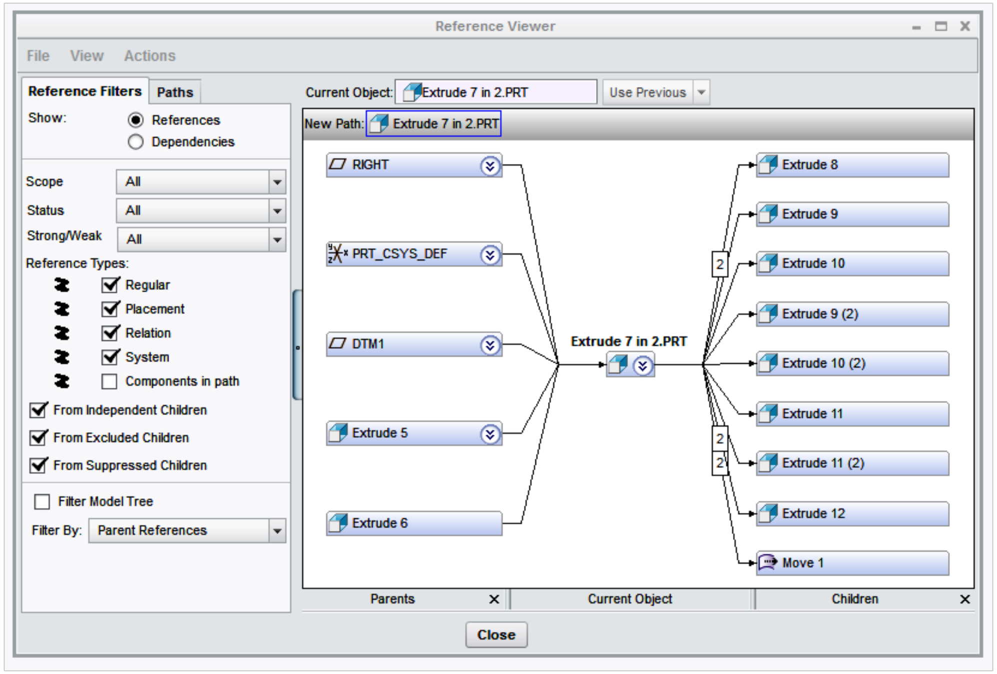
Solidworks

1. High Quality Modeling

Strategies

- Improving the understanding of parent-child relationships

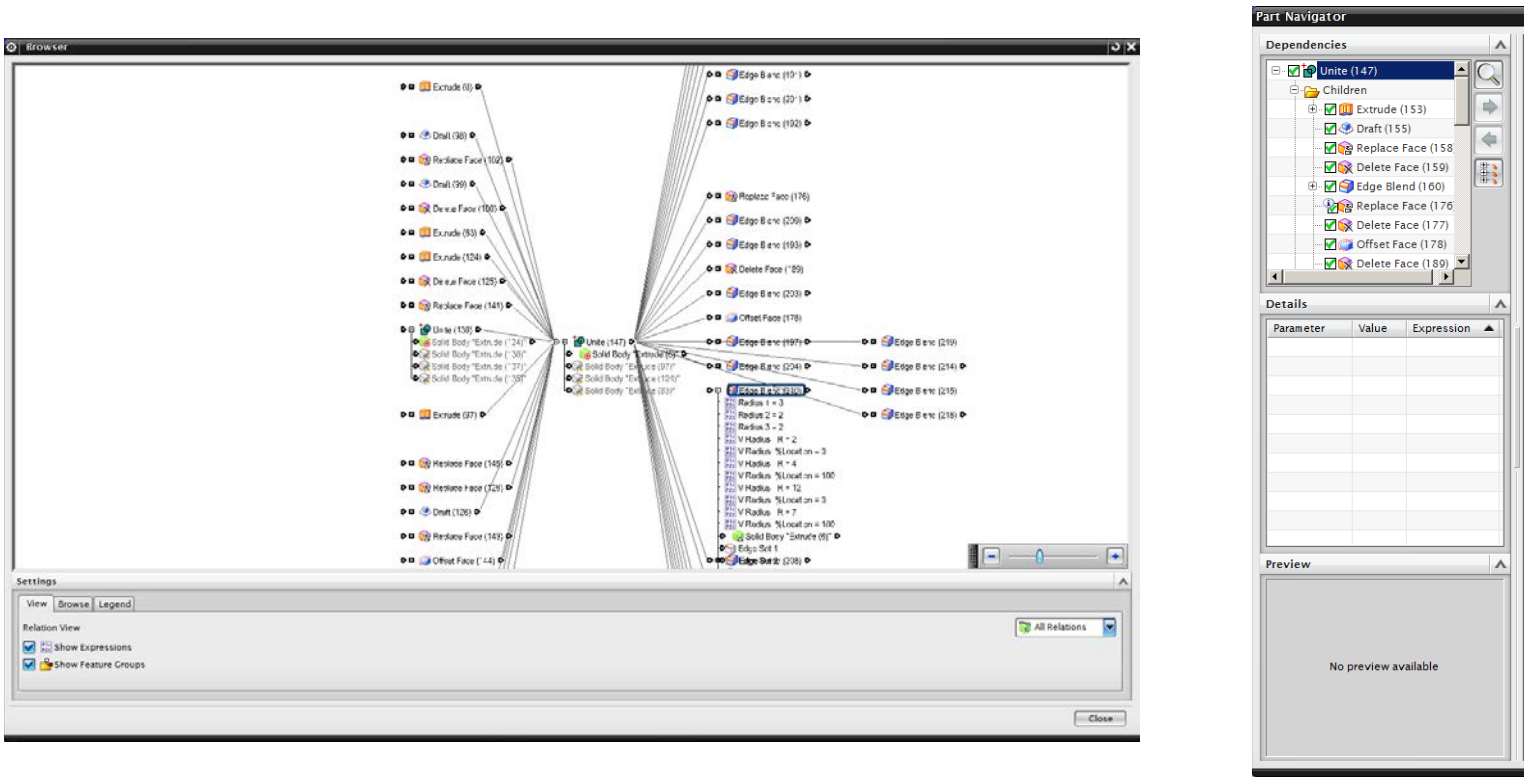




PTC Creo

Strategies

- Improving the understanding of parent-child relationships

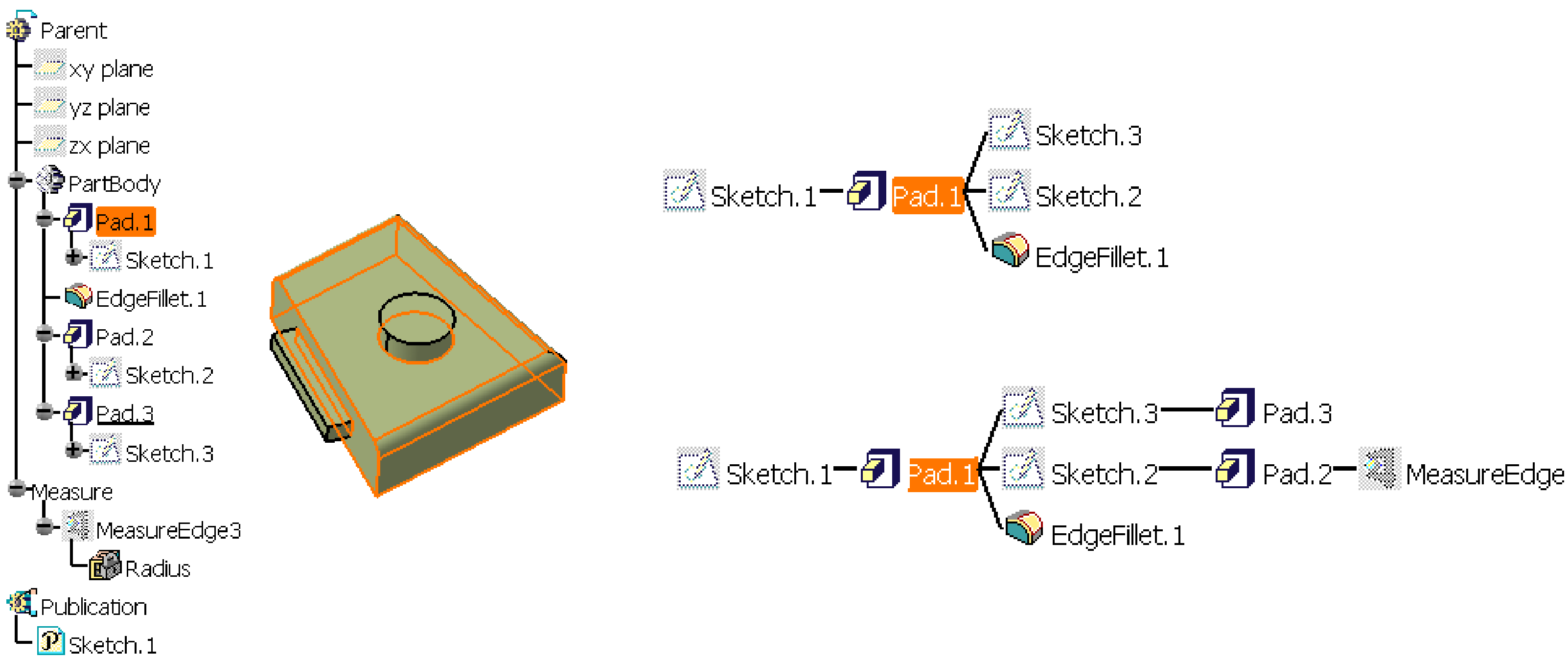


Siemens NX

1. High Quality Modeling

Strategies

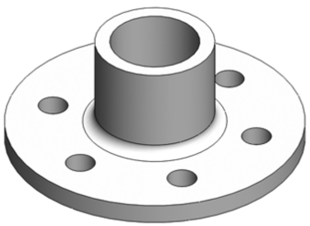
- Improving the understanding of parent-child relationships



CATIA

Strategies

- Improving the understanding of parent-child relationships



Flange
├─ REVOLVE
│ └ *Sketch1*
├─ CUT
│ └ *Sketch 2*
├─ CIR. PATTERN
└─ FILLET

Design Tree

Parent \ Child	Revolve	Cut	Cir. Pattern	Fillet
Revolve		1	0	1
Cut	1		1	0
Cir. Pattern	0	1		0
Fillet	1	0	0	

Dependency Matrix

1. High Quality Modeling

Strategies

- Modeling methodologies

1. High Quality Modeling

Strategies

- Modeling methodologies

1. High Quality Modeling

Strategies

- Modeling methodologies

Horizontally-Structured CAD/CAM Modeling

Landers, D.M., Khurana, P., (2004). *Horizontally-Structured CAD/CAM Modeling for Virtual Concurrent Product and Process Design*. US Patent 6,775,581. Delphi Technologies

Explicit Reference Modeling

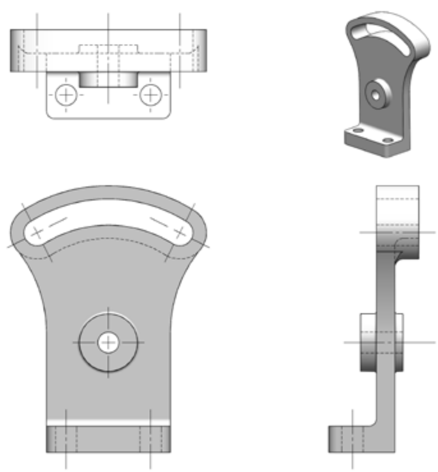
Bodein, Y., Rose, B., Caillaud, E., (2014). *Explicit Reference Modeling Methodology in Parametric CAD System*. Computers in Industry 65(1), pp. 136-147.

Resilient Modeling

Gebhard, R., (2013). *A Resilient Modeling Strategy*. Technical Presentation, Solid Edge University. http://resilientmodeling.com

1. High Quality Modeling

- Design trees by methodology



Horizontal Modeling

- Base Feature 0
- Base 1
- Reference Plane 2
- Reference Plane 3
- Face Plane 4
- Face Plane 5
- Face Plane 6
- Face Plane 7
- Face Plane 8
- Face Plane 9
- Boss-Extrude2
- Boss-Extrude3
- Boss-Extrude4
- Boss-Extrude5
- Fillet1
- Fillet2
- Fillet3
- Fillet4

Expl. Ref. Modeling

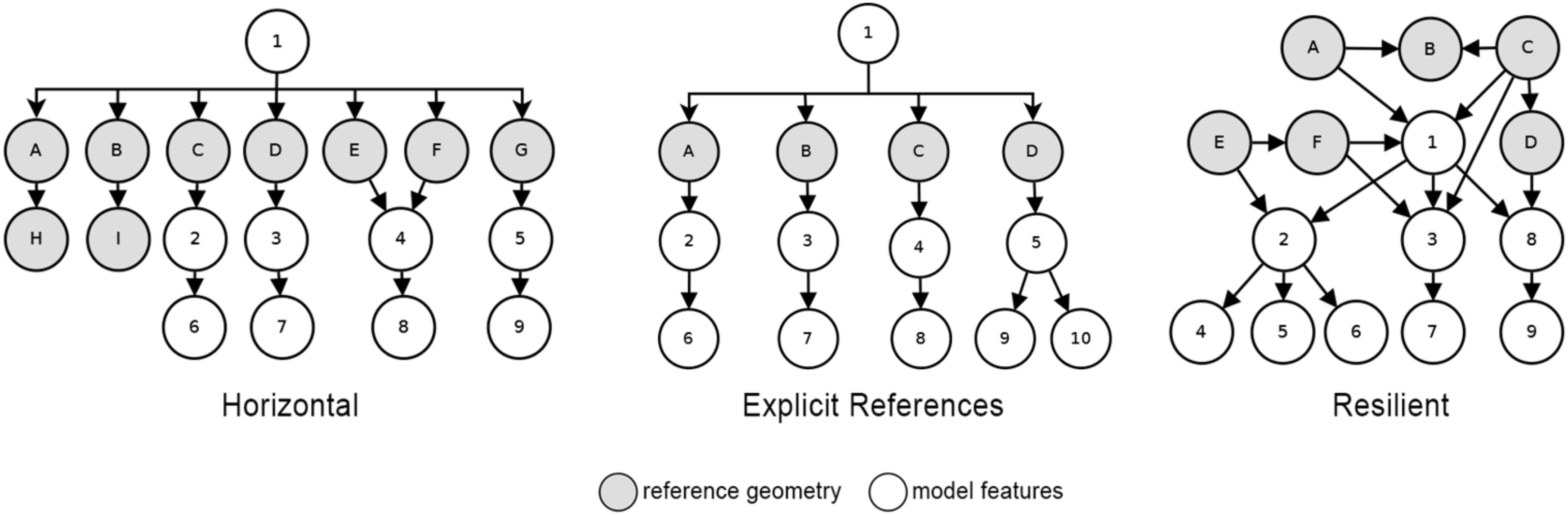
- Plane1
- Plane2
- Plane3
- Plane4
- Boss-Extrude1
- Boss-Extrude2
- Fillet1
- Boss-Extrude3
- Fillet2
- Boss-Extrude4
- Fillet3
- Boss-Extrude5
- Fillet4
- Cut-Extrude1

Resilient Modeling

- Ref
  - Reference Plane 1
  - Reference Plane 2
  - Reference Plane 3
- Construction
  - Plane1
  - Center References
  - Symmetry Axis
- Core
  - Main Body 7mm
  - UpperBody 16mm
  - LowerBody 16x50x10
- Details
  - Crank Support_R4
  - Holes Base_DIA8
- Quarantine
  - Fillet 1mm-Crank Support
  - Fillet 3mm-Lower Corners
  - Fillet 3mm-Upper Body
  - Fillet 3mm-Main and Lower

1. High Quality Modeling

- Graph complexity and structure varies by methodology



1. High Quality Modeling

Contributions

- We have conducted numerous experiments to test and compare different modeling approaches in a variety of design scenarios and modeling situations .
- Levels of quality (Linguistic Model):
  - Morphological, syntactical, and semantic.
- Dimensions of quality:
  - Validity, completeness, consistency, conciseness, clarity, and design intent.
- The goal is to develop a defined set of quantitative metrics.

2. Model Quality Testing

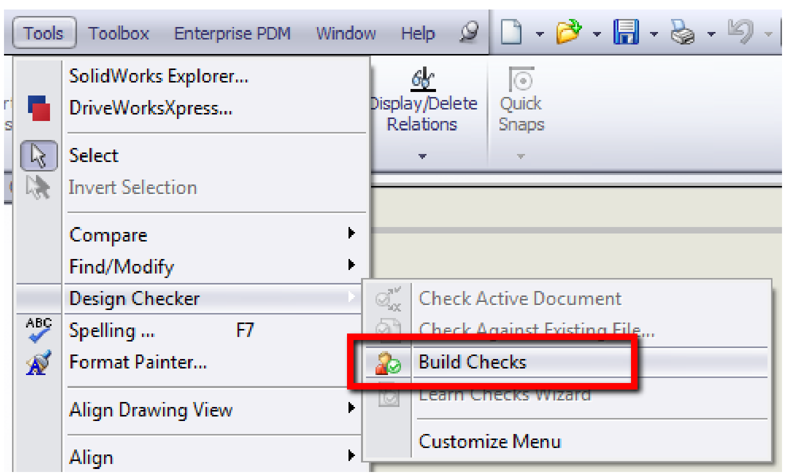
© Roger Penwill

2. Model Quality Testing

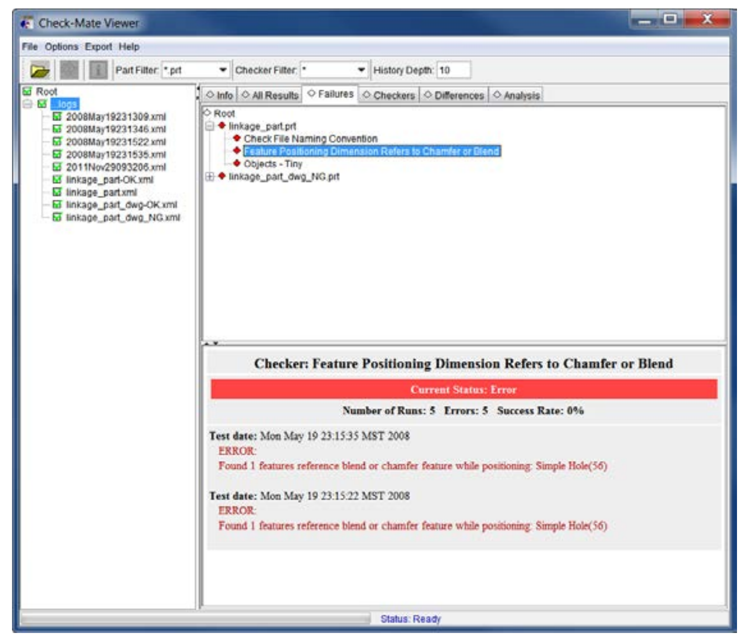
Analyze, correct, and repair defects (particularly at the semantic level) is still necessary to guarantee the reliability and robustness of master digital models.

Model Quality Testing tools

- Quality testing tools for model verification, validation, and comparison are essential, as exporting CAD models that contain errors or anomalies to different downstream applications is prone to data corruption, which typically requires the models to be reworked by the downstream user.



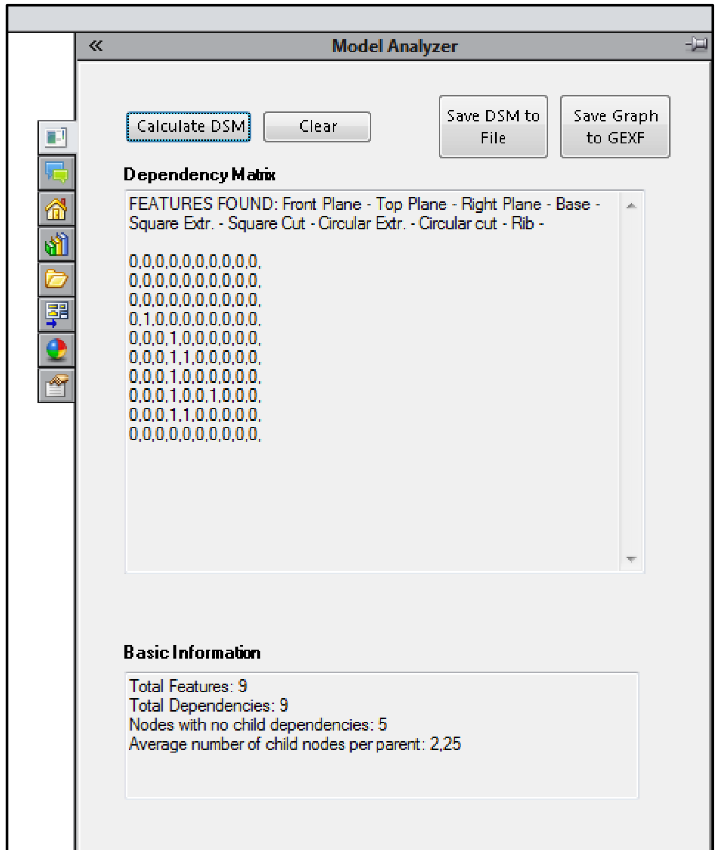
SolidWorks Design Checker



NX Checkmate

Model Quality Testing Contributions

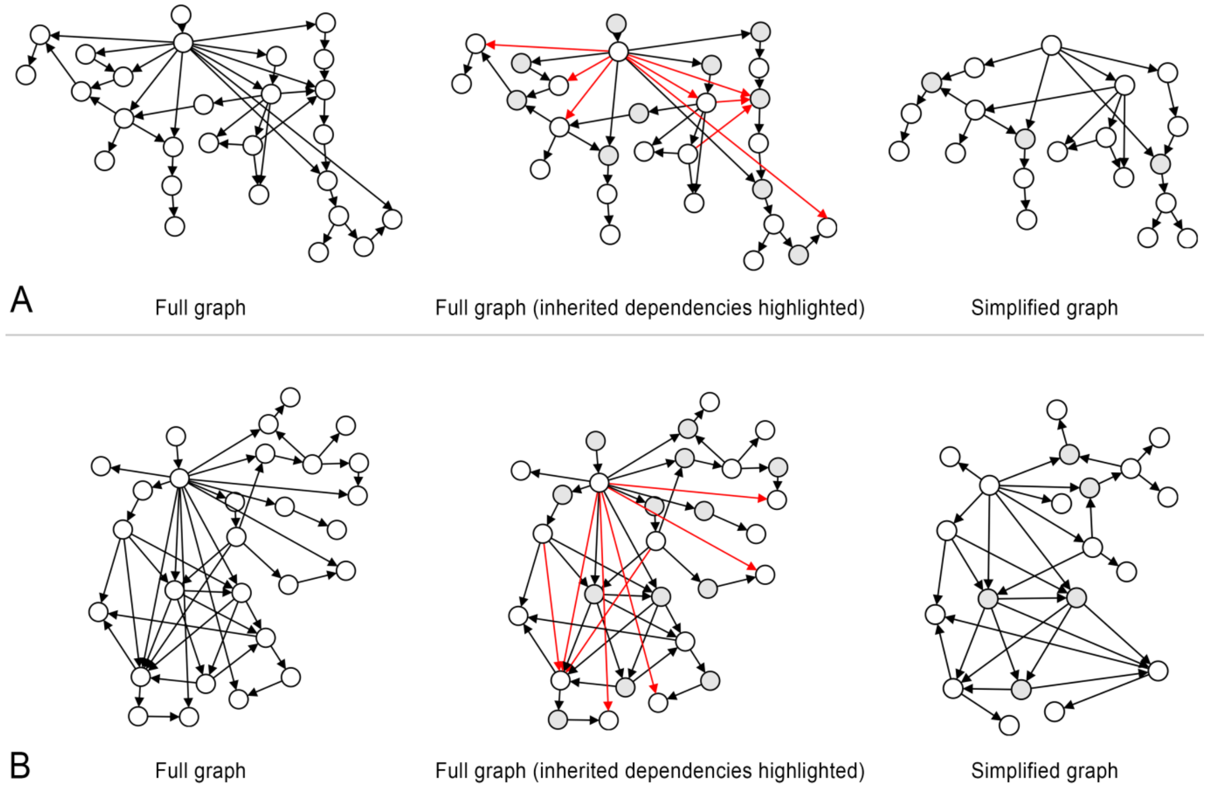
- DSM Model analyzer for Solidworks.



2. Model Quality Testing

Model Quality Testing Contributions

- Management of inherited dependencies in DAG.



2. Model Quality Testing

Model Quality Testing Contributions

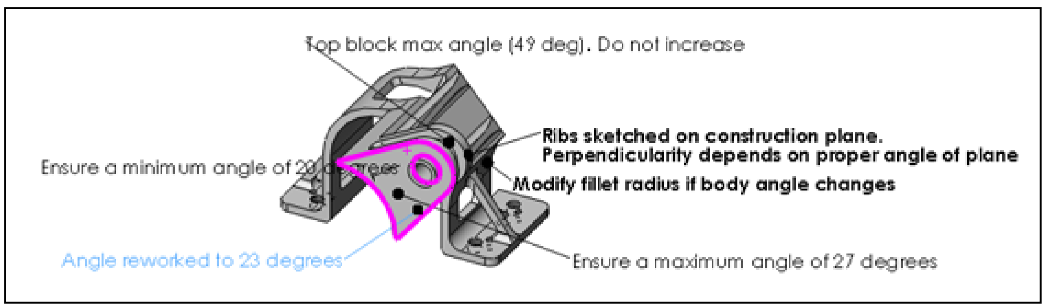
- Detection of Dimensions of Quality.
- Automatic Repair and Correction.

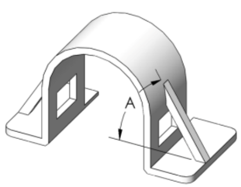
3. Model Enrichment

- Enrich models with structured design information and mechanisms to centralize knowledge and facilitate design communication.

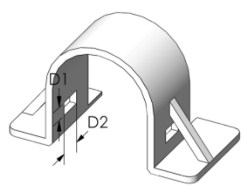
- Annotated models provide significant benefit over non-annotated models in terms of reusability, especially when undergoing design alteration.

- Built-in historical record of design information.

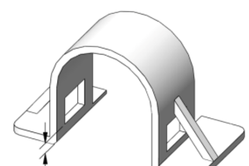




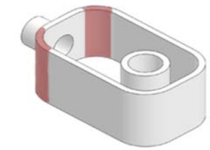
Solution 1: Increase rib angle

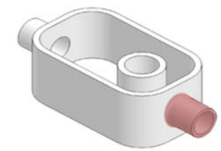


Solution 2: Change side cut dimensions

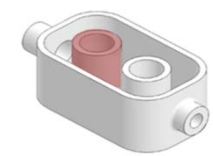


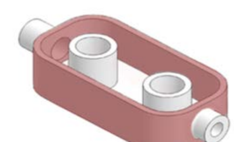
Solution 3. Move side cut down


ALTERATION 1
(fillet edges)


ALTERATION 2
(create outlet)


ALTERATION 3
(modify outlet)


ALTERATION 4
(create new connector)


ALTERATION 5
(modify width and height)

3. Model Enrichment

Challenges

- *Representation:* external vs. internal.
- *Content and Structure:* what needs to be provided and in what form.
- *Interface:* interaction system used to enter and retrieve annotation information.
- *Visualization:* visual clutter, information overload, etc
- *User Motivation:* incentives, documenting, etc.

3. Model Enrichment

Representation

- *Internal:* annotations stored within the CAD model
- *External:* annotations stored in external repositories

	Advantages	Disadvantages
Internal	• Easy implementation • Full integration with the model (low maintenance) • Efficiency in terms of processing and manipulation • Already supported by most CAD systems	• Original document changed • Difficulty for multiple sets of markups • Difficult to search annotations content
External	• Multiple independent sets of markup • Progressively information update • Easy distribution of information in collaborative environments and over the web. • Information can be processed and analyzed separately	• Difficulty of implementation • Problem of persistent references • File maintenance

3. Model Enrichment

Content and Structure

- What information should be included? How do we capture it?
- Structured vs. unstructured
- When is the user providing too much information?
- Can we develop a standard structure or language for annotations?

3. Model Enrichment

Interface

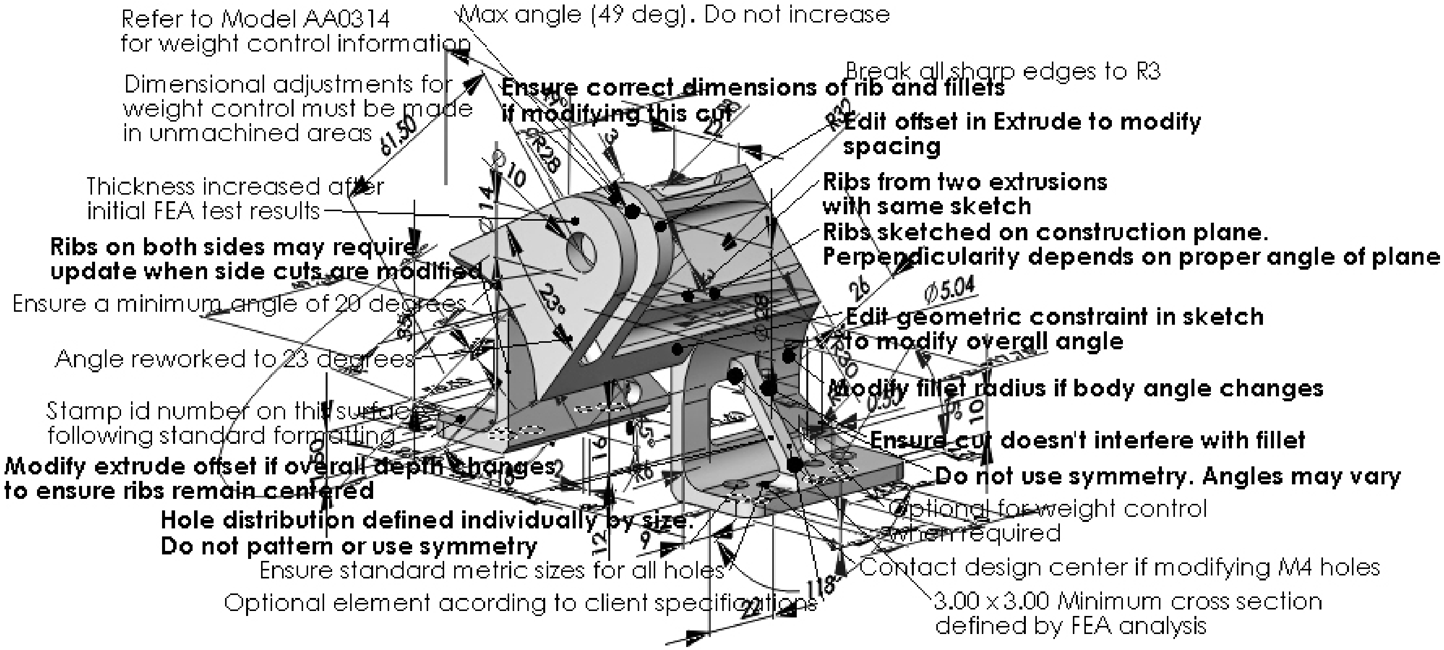
- Mechanisms to support user interaction with the information are needed.
- None of the current standards provide explicit guidelines about managing 3D annotations, although they do recommend the use of groups, layers, or views to make the model more readable.
- Interaction with annotations should be integrated with the CAD application (PMI modules) so designers do not need to leave the CAD environment to annotate the models.
- Effort (time) vs. Benefits

3. Model Enrichment

Interface in current systems

- Current systems: poor usability. Basic information grouping tools.
- No selective visualization.
- No filtering and/or search functionality.
- Disconnected from collaborative design tools.
- No traceability of information and how it evolves over time.
- No links with external documents and information systems.

Visualization



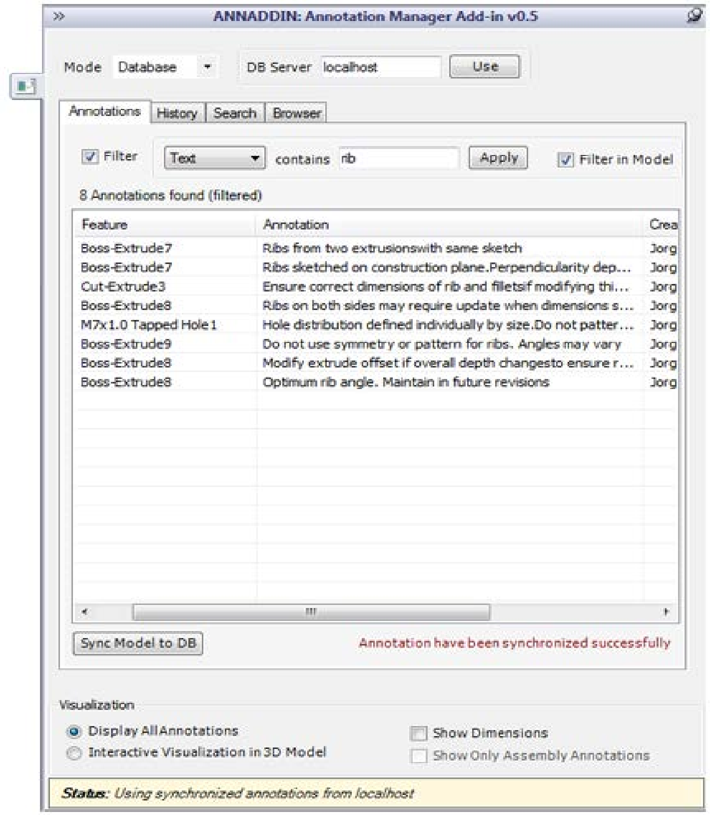
3. Model Enrichment

User Motivation

- Designers are reluctant to spend time annotating their designs
- The designer has no further use of annotations, as she already understands the design. Why should the designer do something that is only beneficial for people that come after her?

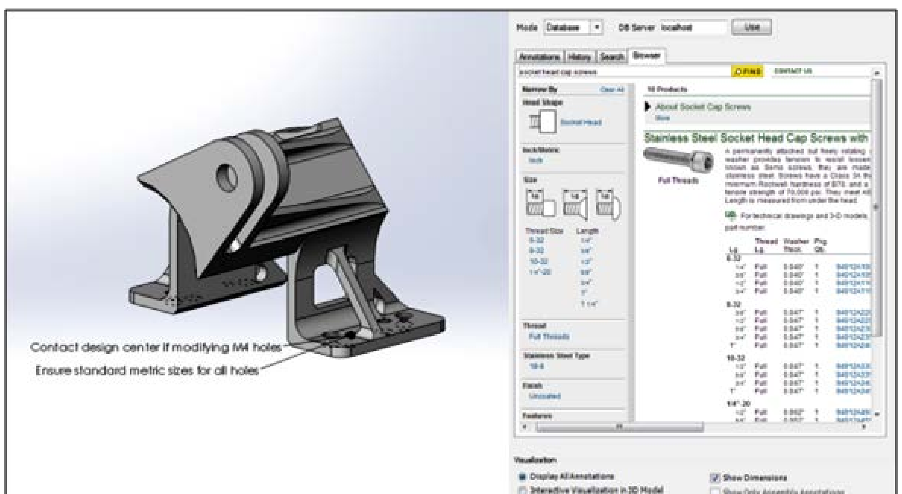
Contributions: Annotation Manager

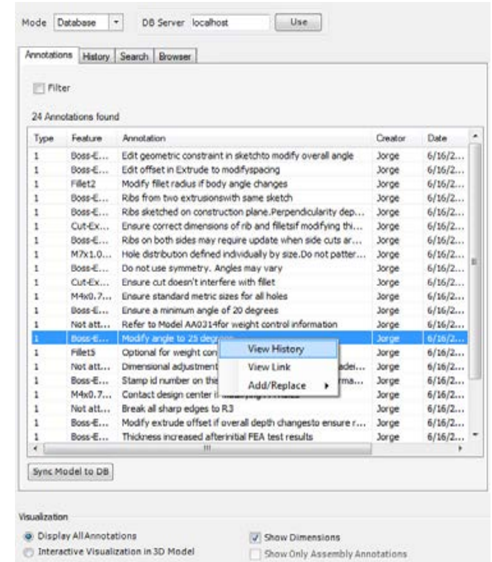
- CAD Integration
- Enhanced User Interface
- Advanced visualization, filtering and search functionalities
- Interactive visualization via model features

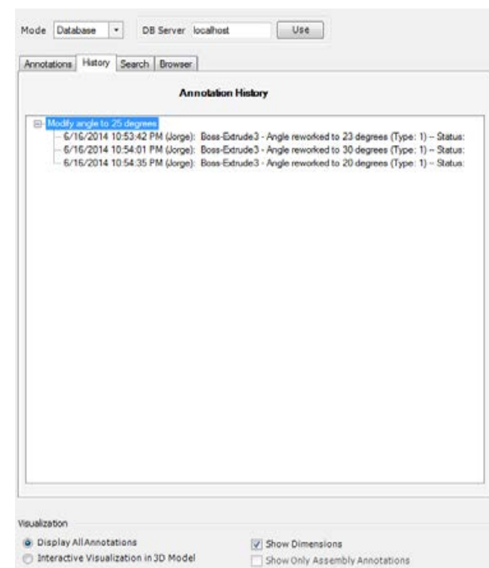


Contributions: Annotation Manager

- External document support (hyperlinks, multimedia content, etc.)
- PLM Integration
- Historical records and information evolution
- Videoconferencing

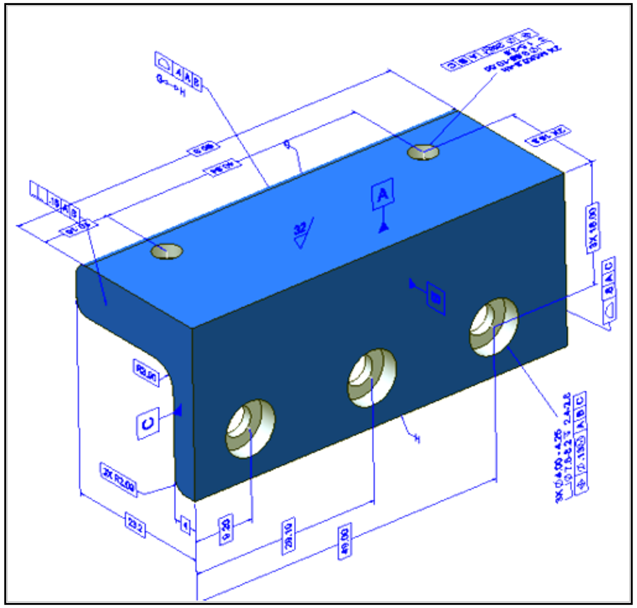






3. Model Enrichment

Demo



Thank You

Jorge D. Camba
jdorribo@uh.edu
UNIVERSITY of HOUSTON

Manuel Contero
mcontero@upv.es
UNIVERSIDAD POLITECNICA DE VALENCIA

Pedro Company
pcompany@uji.es
UNIVERSITAT JAUME I

The icons used in this presentation are courtesy of José Sanmartín González, Delwar Hossain, Justin Blake, Hea Poh Lin, and Creative Stall from Noun Project (https://thenounproject.com)