PUBLICLY AVAILABLE SPECIFICATION

ISO/PAS 26183

First edition
2006-09-01

SASIG Product data quality guidelines for the global automotive industry

Principes directeurs SASIG relatifs à la qualité des données de produit pour l'industrie automobile mondiale

Reference number
ISO/PAS 26183:2006(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

In other circumstances, particularly when there is an urgent market requirement for such documents, a technical committee may decide to publish other types of normative document:

— an ISO Publicly Available Specification (ISO/PAS) represents an agreement between technical experts in an ISO working group and is accepted for publication if it is approved by more than 50 % of the members of the parent committee casting a vote;

— an ISO Technical Specification (ISO/TS) represents an agreement between the members of a technical committee and is accepted for publication if it is approved by 2/3 of the members of the committee casting a vote.

An ISO/PAS or ISO/TS is reviewed after three years in order to decide whether it will be confirmed for a further three years, revised to become an International Standard, or withdrawn. If the ISO/PAS or ISO/TS is confirmed, it is reviewed again after a further three years, at which time it must either be transformed into an International Standard or be withdrawn.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO/PAS 26183 was transposed by Technical Committee ISO/TC184, *Industrial automation systems and integration*, Subcommittee SC4, *Industrial data*, in collaboration with the Strategic Automotive product data Standards Industry Group (SASIG) under the ISO/TC184/SC4 Harvesting Process, as defined in SC4 Standing Document (SC4N1198), *Procedures for Transposing Externally Developed Specifications into ISO Deliverables*.

At the time this document was created, SASIG member organizations included the following:

— Automotive Industry Action Group (U.S.)

— Federal Chamber of Automotive Industries (Australia)

— Groupement pour l'Amélioration des Liaisons dans l'Industrie Automobile (France)

— Japan Automotive Manufacturers Association (Japan)

— ODETTE Sweden

— Verband der Automobilindustrie (Germany)

This set of guidelines is a joint effort by the organizations that comprise SASIG. In particular, the following organizations contributed the major content of this work:

— Automotive Industry Action Group (AIAG)

— Groupement pour l'Amélioration des Liaisons dans l'Industrie Automobile (GALIA)

— Japan Automotive Manufacturers Association (JAMA)

— Verband der Automobilindustrie (VDA)

— ODETTE Sweden

Other SASIG member organizations that contributed to the document include:

— Federal Chamber of Automotive Industries (FCAI)

Though not SASIG member organizations, the Japan Automobile Parts Industry Association (JAPIA) and the Verband Deutscher Maschinen- und Anlagenbau (VDMA) also contributed to the guidelines.

SASIG

Product Data Quality

Guidelines for the Global Automotive Industry

Document Version 2 Revision 1, May 2005

(replaces Document Version 2.0, September 2004)

SASIG

Automotive Industry Action Group (U.S.), Groupement pour l'Amelioration des Liaisons dans l'Industrie Automobile (France), Japan Automobile Manufacturers Association (Japan) / Japan Auto Parts Industries Association (Japan), Odette Sweden AB (Sweden), and Verband der Automobilindustrie (Germany) are members of the Strategic Automotive product data Standards Industry Group- SASIG. These organisations are cooperating in the creation, distribution and use of joint documents, including PDQ (Product Data Quality), PDM Assembly Data Exchange, XMTD (Exchange and Management of Technical Data), DEV (Digital Engineering Visualization).

SASIG PUBLICATIONS

A SASIG publication reflects a consensus of those substantially concerned with its scope and provisions. A SASIG publication is intended as a guide to aid the manufacturer, the consumer and the general public. The existence of a SASIG publication does not in any respect preclude anyone from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the publication.

CAUTIONARY NOTICE

SASIG publications are subject to periodic review and users are cautioned to obtain the latest editions.

MAINTENANCE PROCEDURE

Recognizing that this SASIG publication may not cover all circumstances, SASIG has established a maintenance procedure. Please refer to the Maintenance Request Form at the back of this document to submit a request.

Contact information

Contacts information to SASIG is available on http://www.sasig.com

SASIG Copyright and Trademark Notice:

The contents of all published materials are copyrighted by members of SASIG, AIAG, Galia, JAMA/JAPIA, Odette Sweden, VDA Copyright is not claimed as to any part of an original work prepared.

CHANGE HISTORY V1.0 - V1.1, FEBRUARY 2003

ALL SECTIONS:

General editing and clean-up of terms, references and graphic samples.

SECTION I:

Add Criteria numbers to Table 5, rotate 90^0 and enlarge print for readability.

SECTION II:

Criteria 3.2.3.1 Change Non-Nurb to Analytical, all uses.

Criteria 3.2.1.12 inappropriate degree linear curve: G-CU-ID, added

Criteria 3.2.2.17 Multi-faced surface: G-SU-MU, added.

Criteria 3.2.2.17 Folded surface; G-SU-FO, deleted.

Criteria 3.2.5.2 Encoding modified.

Criteria 3.2.5.3, Non-nurbs face changed to Analytical Face; G-FA-AN.

Criteria 3.2.5.7, Multi-region face: G-FA-MU deleted.

Criteria 3.3.2 Identical elements changed to Embedded elements.

ATTACHMENTS:

Attachment A : Glossary

Applied approved Glossary changes.

Attachment C : Recommended Values

Table replaced with statement directing the reader to the websites of the respective organisations for recommended values to be used in evaluating model quality.

Attachment D : Formsheet

Changed non-nurb to analytical.

CHANGE HISTORY V1.1 - V2.0, SEPTEMBER 2004

ALL SECTIONS:

General editing and clean-up of terms, references and graphic samples.

Control of English UK usage.

Document reorganisation:

Section III becomes Attachment F (former Attachment F is Attachment G)

Former Section IV becomes Section III

SECTION I:

New chapter 1.7 How to use these guidelines

SECTION II:

Encoding has been modified to be compliant with the criteria reorganisation.

The former chapter 3.1 (Best Practices) has been deleted and its content has been dispatched into the Non-geometric criteria chapter.

Chapter 3.1.2 Surface : Two criteria added :

- "Folded surface"
- "Inappropriate degree planar surface"

Chapter 3.1.5 Face : Numbering correction

Chapter 3.1.7 Solid : The three first former criteria has been transferred to the Non-geometric criteria chapter.

Chapter 3.2.8 Model : The content has been transferred to the Non-geometric criteria chapter.

Non-geometric criteria chapter has been moved after the Geometric criteria chapter,and has been completed.

Drawing Quality Criteria chapter : twelve new criteria.

Chapter 4 CAE Data : Completed with thirteen families of criteria.

Chapter 9 Quality Stamp : new chapter.

Former chapter 9 Other Data becomes chapter 10.

SECTION III:

Chapter 11.11 Healing has been completed.

ATTACHMENTS:

Attachment A : Glossary
Unused words have been deleted.

Attachment B : Mapping between element types
Columns of Pro/E have been completed.

Attachment F : Business Case

Attachment G : Revision request

CHANGE HISTORY V2.0 - V2.1, MAY 2005

ALL SECTIONS:

General editing and clean-up of terms, references and graphic samples.

Control of English UK usage.

FOREWORD:

Paragraph regarding improvements updated.

SECTION I:

-

SECTION II:

Chapter 9 Quality stamp has been revised.

SECTION III:

-

ATTACHMENTS:

Attachment A : Glossary
New words have been added.

Attachment B : Mapping between element types
Columns of CATIA V4 and CADCEUS have been revised.
Column for CATIA V5 has been added.

Attachment H : XML Schema for the quality stamp has been added

Attachment I : XML File Example has been added

TABLE OF CONTENTS

Overview

Detailed Contents

Figures

Tables

EXECUTIVE SUMMARY

The global automotive industry is increasingly dependent on electronic product data to design and produce vehicles. Because of that dependency, problems with the quality of product data lead to problems developing and producing the products. When its member organisations recognised the problem, the Strategic Automotive product data Standards Industry Group (SASIG) decided the most effective approach would be to produce this common set of guidelines on the aspects of product data quality.

For the purposes of these guidelines, the term "product data" is defined as any and all product data required from product conception to manufacturing. Therefore, product data include not just computer-aided design (CAD) data but also computer-aided manufacturing (CAM) data, computer-aided engineering (CAE) data, product data management (PDM) data, and other kinds of data.

Following is the definition of product data quality, on which this set of guidelines is based:

> ***Product data quality*** is a measure of the accuracy and appropriateness of product data combined with the timeliness with which those data are provided to all the people who need them.

From this we can state that: Good product data quality means providing the *right data* to the *right people* at the *right time.*

During product development, many people depend on data describing various aspects of the product. The need for high quality product data is easy to describe at a high level: poor quality data costs money, delays product development, and can result in poor quality products. These entire costs can be increased even more by the need to spend extra money to meet a product development schedule and pay for overtime labour or bring in temporary contract personnel to assist. Unfortunately, connecting PDQ costs to their causes is rarely simple.

This set of guidelines provides a broad range of information captured in three main sections.

- Section I provides introductory and background material that frames the product data quality problem. Topics covered include the nature of product data, high-level product data quality issues, and how to use •this document.
- Section II contains specific product data quality criteria for users. In this version, the content focuses primarily on CAD geometry, though other topics are also at least partially addressed. The criteria describe specific problems that can occur and suggest how to measure them and what to do when they occur.
- Section III provides information and methods that will help improve product data quality. The topics covered range from readiness for change to reward systems to supporting technologies such as tools for checking data.

In addition to the main parts, a set of attachments provides further information.

This document can be considered a work in progress. SASIG chose to publish at this stage because delaying to fill in the gaps would keep valuable information out of the hands of those who need it.

Direct use of these guidelines by product data creators and users of all types is encouraged. The people most likely to make regular use of Section II of these guidelines will be those who are responsible for building and maintaining company product data guidelines or standards. Those most likely to use Sections III and Attachment F are the managers, team leaders, and other people responsible for business processes and results.

FOREWORD

The Strategic Automotive product data Standards Industry Group (SASIG) comprises automotive industry organisations from around the world. It was originally formed in 1994 to encourage the development and promotion of STEP, the international product data exchange standard (ISO 10303), within the automotive industry. Individually, the SASIG member organisations realised that if the quality of the product data being exchanged was poor, even the best data exchange processes would be of little value. They also realised that problems with product data quality were widespread and costly. Each organisation was, therefore, developing its own guidelines and recommendations for improving product data quality.

In 1999, the SASIG member organisations recognised their common interest in product data quality. Verband der Automobilindustrie (VDA) proposed a focused meeting to consider how SASIG could encourage greater collaboration in addressing this common issue. At that special meeting, the organisations decided to cooperate in developing a common set of guidelines based on VDA's existing CAD data quality guideline (VDA 4955 V2) and other organisations' documents. The intended result would encourage the most effective and broadly applicable product data quality for the global automotive industry. This commitment led to a series of workshops which, in turn, led to the collaborative development of this set of guidelines.

This is Version 2 of the product data quality guidelines. As such, it does not cover all the potential product data quality issues that have been identified. In the interest of putting useful information in the hands of those who can use it, the participating organisations have decided to move forward with Version 2 by gathering information already developed. The expected additional major topics for future versions are included as major sections with summary information about what will eventually be covered.

Revisions of the documents are published in case that corrections improve the quality of the documents itself. SASIG and its Product Data Quality Work Group encourage suggestions for improving this set of guidelines. Please use the form given with Attachment G.

SASIG encourages participation in this and its other activities by national automotive organisations from all countries.

Disclaimer and Explanation

At the time this document was created, Strategic Automotive product data Standards Industry Group member organisations included the following:

- Automotive Industry Action Group (U.S.)
- Federal Chamber of Automotive Industries (Australia)
- Groupement pour l'Amélioration des Liaisons dans l'Industrie Automobile (France)
- Japan Automotive Manufacturers Association (Japan)
- ODETTE Sweden
- Verband der Automobilindustrie (Germany)

The SASIG member organisations are expected to distribute these guidelines to their respective members and other interested parties in their countries. While the SASIG member organisations are free to translate these guidelines into other languages, this English version shall always be considered the master document in case of a dispute. While distribution of the guidelines will normally be done as a complete document, the SASIG member organisations are free to extract portions for use in other documents. In such cases, the original source of the information shall be clearly documented. SASIG member organisations may also distribute the guidelines through the SASIG PDQ Liaison Member organisations.

ACKNOWLEDGEMENTS

This set of guidelines is a joint effort by the organisations that comprise SASIG. In particular, the following organisations contributed the major content of this work:

- Automotive Industry Action Group (AIAG)
- Groupement pour l'Amélioration des Liaisons dans l'Industrie Automobile (GALIA)
- Japan Automotive Manufacturers Association (JAMA)
- Verband der Automobilindustrie (VDA)
- ODETTE Sweden

Other SASIG member organisations that contributed to the document include:

- Federal Chamber of Automotive Industries (FCAI)

Though not SASIG member organisations, the Japan Automobile Parts Industry Association (JAPIA) and the Verband Deutscher Maschinen-und Anlagenbau (VDMA) also contributed to the guidelines. Direct contributors to the document include the following:

Participant	Company
AIAG	
Jerry Brooks	Ford Motor Company
James Bullard	General Motors
W. Kirk Crawford	DaimlerChrysler
Don Galway	Dana Corp.
Lori Kell	General Motors
Jeff C. Kitson	General Motors
Philip Marlow	Visteon
GALIA	
Frédéric Chambolle	PSA Peugeot Citroën
François Desnoyer	Renault
Philippe du Rivault	GALIA
Christine Fievet	Valeo
Thomas Fischer	Robert Bosch France S.A
Louis Fort	Renault
Michel Genouille	Usinor
Pierre Germain-Lacour	PSA Peugeot Citroën
Christian Jacquot	Faurecia
Rida Khatir	PSA Peugeot Citroën
Bernard Osiowski	Faurecia
Mario Picco	CETIM
Oscar Rocha	Renault
JAMA and JAPIA	
Noriyuki Hiratsuka	Mitsubishi Motors
Masato Ichioka	Denso i-tech
Yoichi Ieda	Toyoda-gosei
Hiroki Ifuku	MMC Computer Reserch
Masazumi Itoh	CalsonicKansei
Hidetaka Motooka	Stanley Electric
Masanobu Nakamura	Mitsubishi Automotive Engineering
Toshihiro Nakayama	Nissan Motor
Mitsuo Ogata	Stanley Electric
Mari Omura	Fujitsu Kyushu System Engineering
Hidenobu Shimizu	Denso i-tech
Akira Suzuki	Toyota Motor
Kazuharu Taga	Honda Engineering
Toshiyuki Tomita	Toyoda-gosei
Toshiya Yoshitani	Mitsubishi Motors
ODETTE SWEDEN	
Roger Andersson	Chalmers
Sven Andersson	IVF
Bo Johansson	Chalmers
VDA	
Silvia Beutler	DaimlerChrysler AG
Andreas Groll	Karmann GmbH
Hans-Jürgen Hochfeld	Volkswagen AG
Rudolf Hummel	ZF AG
Andrea Knaute-Haase	Robert Bosch GmbH
Julia Litterscheidt	BMW AG
Hartmut Steins	Pierburg AG
Lutz Völkerath	PROSTEP AG

(Blank page)

SECTION I: INTRODUCTION AND BACKGROUND

This Section introduces product data quality and provides background that frames the rest of the document. It also provides guidance on using these guidelines.

1 Introduction

The global automotive industry is increasingly dependent on electronic product data to design and produce vehicles. Because of that dependency, problems with the quality of product data cause problems in developing and producing the products. When its member organisations all recognised the problem and were all working on some aspects of product data quality (PDQ), SASIG decided that the most effective approach would be to produce this common set of guidelines on the aspects of PDQ. Addressing product data quality is a complicated issue. One underlying problem is that each user of product data has different requirements for that data. The following are a few examples that focus on CAD data.

- *CNC programming* – A CAD model of a part or tool that looks good on the screen or when printed on paper may not contain the right set of information necessary for automatically programming a CNC machine tool to make the part. There may be excess geometry or incomplete geometry, either of which confuse the CNC programming software or result in an incorrect part or tool.
- *Structural finite element analysis* – A CAD model that represents a detailed design typically has far too much detail for structural analysis. The model needs to be simplified for a realistic chance of getting a useful simulation. Examples of details that can get in the way include fillets and radii added for such purposes as making the part more manufacturable.
- *Tooling design* – The original CAD model for a part that is going to be moulded or cast may not have the parting line identified or the necessary draft built in. The allowance for the inevitable shrinkage of a solidifying part may not be built into the model either. Similar allowances have to be built into stamping dies for over-bending to compensate for spring back.

When too much or too little information is provided or the information is incorrect, the result is increased cost and time. To be able to use poor quality data, someone must spend time, often extensive, putting the model in a usable form. Sometimes the required changes are so time-consuming and expensive that the recreation of a correct version of the model is more cost-effective. Recreating data leads to the potential for introducing further errors in the data.

1.1 Product Data

For the purposes of this set of guidelines, the term “product data” is defined as any and all product data required from product conception to manufacturing. Therefore, product data includes not just computer-aided design (CAD) data but also computer-aided manufacturing (CAM) data, computer-aided engineering (CAE) data, product data management (PDM) data, and other kinds of data. Examples of product data include:

- CAD models (solid, surface, or wireframe)
- Engineering and manufacturing bills of materials
- Process plans
- Model revision history and effectivity
- Product assembly structure
- Numerically controlled machine-tool programs

1.2 Define PDQ

The best way to characterise product data quality involves a base statement with two implicit themes. The definition of product data quality on which this set of guidelines is based is:

> ***Product data quality*** is a measure of the accuracy and appropriateness of product data combined with the timeliness with which those data are provided to all the people who need them.
>
> From this we can state that:
>
> Good product data quality means providing the *right data* to the *right people* at the *right time.*

The first theme implicit in this definition is the need for an appropriate set of metrics. To improve product data quality, one must be able to measure the level of product data quality and, after making a change, evaluate whether an improvement has occurred. Section II has some recommendations on metrics.

The second implicit theme is access. Regardless of how appropriate and robust a specific product model or set of data is, if it is not available in a timely manner to those needing it, then that model or data set is of no value. Examples of this include inappropriate data formats or systems, denied access to file servers, missing part numbers, and hidden data files. Issues that deal with how data are accessed fall in the realm of PDM systems. The primary access issues are who has access to what and when. Of course, the data created and maintained by PDM systems are also product data. Hence, those data are also subject to product data quality concerns. This view is driven by the requirements for concurrent engineering. The underlying principle of concurrency is that downstream activities start before upstream activities are complete. This principle requires consideration of which data are needed, at what time, and by whom. Thus, the information about where those models or bills of material are to be found as well as how and when to get them can, and should, also be considered product data.

1.3 Need for PDQ

The need for high quality product data is easy to describe at a high level: poor data quality costs money, delays product development, and can result in poor quality products. Unfortunately, connecting PDQ costs to their causes is generally not so simple. During product development, many people depend on data describing various aspects of the product, including:

- CAD users, who need data to help them develop related parts or other aspects of the product based (in part, at least) on the geometry.
- Engineering analysis specialists, who need data to build analysis models.
- Tooling and fixturing designers, who need accurate part geometry as a basis for the design of their manufacturing equipment.
- Numerically controlled machine tool programmers, who need to develop the program used to machine parts.
- Prototype builders, who need good representation of the products they have to make.
- Product quality inspectors, who need accurate representations of parts to ensure that they meet the design requirements.
- Testing laboratories, who must understand the nature of a product in detail before they can complete a test.

Problems in any of these or similar areas can generate substantial costs. The costs can be directly realised by the need to:

- Check data for problems, regularly and repeatedly.
- When they are discovered, try to fix (successfully or not) data problems that are discovered.

- Spend time re-entering problem data into a different system.
- Resend corrected data.
- Resend unsuccessfully or incompletely received data.

In addition, there can be indirect costs due to poor quality product data. These costs result from the effect of data problems on work done or products produced using the data, such as:

- Correcting errors that appeared during data re-entry or correction
- Modifying or re-creating tooling
- Fixing warranty problems
- Re-doing analyses
- Re-building prototypes
- Delay in bringing the product to market

All of these costs can be increased even more by the need to spend extra money to meet a product development schedule and by having to pay for overtime labour or bring in temporary contract personnel to assist.

1.4 Master Data

The extensive use of computer-aided design to create geometric part data has given rise to multiple representations for a component being provided to data recipients (for example, 3D solid, 3D wireframe, and 2D electronic drawings). When different representations of the same part do not agree, the recipient must decide which is correct. The lack of decision criteria has provided the opportunity for misinterpretation. The following hierarchy defines which type of data has precedence (overrules the others) in case of disagreement.

1. If a solid model is present, it should be considered the master data. Any other geometric representations that do not agree should be considered to be in error.
2. If there is no solid model, then a surface model, if present, should be considered the master data. Any other geometric representations that do not agree should be considered to be in error.
3. If there is no surface model, then a 3D wireframe, if present, should be considered the master data. Any other geometric representations that do not agree should be considered to be in error.
4. 2D data are to be assumed correct only if they are the only type of data available.

While these rules of hierarchy may cause the use of data that does not agree with the geometry creator's actual intent, having a commonly understood and accepted hierarchy will encourage consistency. When all parties involved understand the hierarchy, then the creator will know which data type must be correct.

Another problem arises from confusion due to too much information, where another hierarchical approach can help. Because what is necessary versus what is considered superfluous is situation-dependent, the way all geometry should be managed and presented must be defined.

Filtering the display of data elements to include only the most complex element types that define the physical attributes of the finished component (i.e., edges, holes, etc.) should be the highest order in the hierarchy. With this filter applied, the display would show the master data. Filters that include elements such as construction geometry, reference geometry, history, etc., would be filters of a lower order. If a conflict then exists in the other element data included in the data file, the non-master data would be considered, by default, to be in error. An example of these conflicts is when a centreline does not pass through the centre of a hole; the hole would be considered the master (see **Figure 1**). The hole is the physical attribute. The centreline is not a physical attribute of the part and would be filtered out by this requirement.

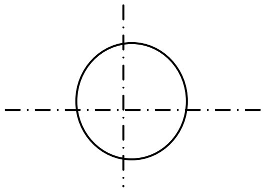


Figure 1. Non-master centrelines incorrectly positioned

Another example is wireframe elements that do not coincide with the edges of a solid. The solid would be considered a more complex element than the wireframe and therefore the solid would be the only element displayed with the required filter applied (see **Figure 2**). The solid would be considered the master and the wireframe would be considered in error.

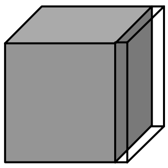


Figure 2. Non-master wireframe structure inconsistent with solid

1.5 Drawing Simplification and Elimination

One goal of any product data quality improvement activity should be to reduce the reliance on paper drawings and increase reliance on electronic representations of the product. This can be accomplished in a phased approach where the content of drawings is simplified and reduced, coupled with increasing reliance on electronic forms of the data. Hence, two companies negotiating CAD issues prior to contract should actively consider simplifying and reducing the content of their drawings. To simplify the overall process, a distinction should be made among **drawings**, **simplified drawings,** and **information drawings**:

Drawing – This term describes the technical drawing in the conventional, classical sense according to drawing standards such as DIN 199 or ISO 5457:1999. The corresponding drawing "represents an object in its intended finished condition;" whereby, "in its depiction and with subsequent details, it takes into account in a certain manner particular view points of the production." This means it is possible to fabricate a product based on the component part drawings. In addition, an assembly drawing shows functional connections and, to a certain extent, the spatial relationship of several component parts to one another. This type of drawing must also satisfy the agreed-upon quality criteria.

Simplified drawing – The contents (scope) of supplementary CAD drawing data may be simplified appropriately given the prerequisite that a complete description of the component part through a 3D CAD data model exists. This simplified CAD drawing data, together with the associated 3D CAD data model, represent the complete data model. Taken together, the two forms of data must be adequate to produce the relevant component part. Both these types of descriptive forms must satisfy the agreed-upon quality criteria.

Information drawing – Information drawings are incomplete and, perhaps, non-scaled depictions of products or component parts. They only serve the purpose of providing supplemental information and generally do not fall under any release procedures or any change control services. Because they do not contain fundamental data needed for producing a part, information drawings need not undergo any quality checks. For data quality considerations, the drawing used to make an offer is, for example, considered to be an information drawing. Ultimately, all exchange of information, both master and illustrative, should be electronic representations, ideally 3D CAD. That is, no "hard copy" drawings or illustrations are needed.

1.6 PDQ Document Strategy

SASIG has determined that this set of guidelines will be published in a series of versions with increasing content.

Table 1. documents the PDQ strategy for developing the content of the different versions. The cell entries and their meanings are:

- A – The need to address the topic is well established and the PDQ criteria are well-defined.
- B – The need to address the topic is well established but PDQ criteria are not yet well-defined.

Suggestions for the improvement of these guidelines are encouraged through the use of the form provided in Attachment G - Revision request.

Table 1. PDQ document strategy

Content marked A appears in this version Content marked B is expected to appear in later versions				Process Chain							
				Define Product	Style Product	Design Product	Evaluate Product	Plan Production	Design Tool	Manufacture Production Tool	Test Tool to Control Quality
Data Class	CAD data	Geometry	Wireframe Model	A	A	A	A	A	A	A	A
			Surface Model	n/a	A	A	A	A	A	A	A
			BREP Solid Model	n/a	A	A	A	A	A	A	A
			CSG Model	n/a	A	A	A	A	A	A	A
		Drawings	2D Drawing	B	B	A	B	B	A	B	B
			Associative Drawing	n/a	B	B	B	B	B	B	B
		Non-geometry	Presentation	B	B	B	B	B	B	B	B
			Parametrics	n/a	B	B	B	B	B	B	n/a
			Features	n/a	B	B	B	B	B	B	B
			Assembly	n/a	B	B	B	B	B	B	B
			Tolerancing	n/a	B	B	B	B	B	B	B
			Surface Condition	n/a	B	B	B	B	B	B	B
			Material Properties	n/a	B	B	B	B	B	B	B
			Parts Information	B	B	B	B	B	B	B	B
	CAE data		FEA, BEA, …	n/a	n/a	B	B	B	B	n/a	n/a
			KIN	n/a	n/a	B	B	B	B	n/a	n/a
			ROB	n/a	n/a	B	B	B	B	B	B
	PDM data		Configuration Management	B	B	B	B	B	B	B	B
			Product Structure	B	B	B	B	B	B	B	B
			Product Management Data	B	B	B	B	B	B	B	B
	Inspection data		Measure Data	n/a	B	B	B	n/a	B	B	B
	Prototyping data		STL	n/a	B	n/a	B	B	n/a	n/a	n/a
	Manufacturing data		Process Plan	n/a	B	n/a	B	B	B	B	B
			NC	n/a	B	n/a	B	n/a	n/a	B	B
			RC	n/a	n/a	n/a	B	n/a	n/a	B	B
	Other data		System Property	B	B	B	B	B	B	B	B
			Technical Data	B	B	B	B	B	B	B	B

Table 2. Document sections

Section I – Introduction and Background	
1. Introduction	Introduces product data quality and this set of guidelines.
2. Data Applicability	Describes the types of product data this version of the guidelines covers as well as coverage expected in future versions. Section 2 also describes various general aspects of product data and the product data life cycle.
Section II – PDQ Criteria	
3. CAD Data	Provides the data quality guidelines associated with various aspects of CAD data such as solid models, assemblies, surface models, tolerance data, and drawing views.
4. CAE Data	Provides the data quality guidelines related to data used by and created by computer-aided engineering applications such as finite element analysis, kinematics analysis, and dynamic analysis.
5. PDM Data	Provides the data quality guidelines for the data stored in a product data management (PDM) system.
6. Inspection Data	Provides the data quality guidelines for the data created for, used, and stored in an inspection system.
7. Prototyping Data	Provides the data quality guidelines for the data created, used, and stored in a prototyping system.
8. Manufacturing Data	Provides the data quality guidelines for the data created, used, and stored in a manufacturing system.
9. Quality Stamp	Provides the guidelines for the PDQ check result.
10. Other Data	Provides the data quality guidelines for other kinds of data not captured in the previous sections.
SectionIII – Improving PDQ	
11. Improving Product Data Quality	Provides guidance on implementing PDQ improvements, including the use of appropriate tools and business processes in support of the direct creation of data.
Attachments	
A. Glossary	Glossary of terms
B. Mapping Between Element Types	Mapping between element types across various systems and standards
C. Recommended Values	Suggested values to use (by specific PDQ criterion) when checking for problems.
D. Formsheet	Example process and form to use between trading partners to agree on various aspects of the data they will exchange.
E. SASIG-ODETTE Cross Reference	Cross reference between this document and sections covering the same information in the ODETTE version 2 recommendations.
F. Business Case	Describes how to build a business case for addressing product data quality, complete with templates to help gather and process the data.
G. Revision Request	A form to be used to request a revision in this document.

1.7 How to Use These Guidelines

These guidelines provide a broad range of information. The people most likely to use these guidelines will be those who are responsible for building and maintaining company product data guidelines or standards. However, direct use of these guidelines by product data creators and users of all types is encouraged.

For any particular activity that uses product data, it is unlikely that this entire set of guidelines will apply. A person using these guidelines is, therefore, expected to apply all the parts of this set of guidelines that fit the particular circumstances. For example, when building surface models, the guidelines specific to solid models will probably not be of value, but at the least, the guidelines about wireframe models will apply.

In “real life,” the requirements of a particular activity may force a user to violate one or more guidelines. When that happens, the user being aware that he or she is doing so is of value so that potential downstream problems with the data can be identified in advance.

2 Data Applicability

This chapter discusses how data can be applied and used in the product development process. It describes the data life cycle, what happens to the data from initial creation through manufacturing, archival, and re-use on future programs. It then describes data exchange and transfer, including procedures and recommendations. Another topic is the concept of “healing” of data—the automated repair of data problems—as opposed to identifying and preventing problems during the design process.

2.1 Data Life Cycle

Product data can be thought of as the commodity that is created by the different organisations that are involved in developing a product. This implies that data, just as any product, have a "life-cycle." It begins in the product data's infancy at some conceptual phase then evolves through more robust design and release phases. The data may go through analysis phase, and will repeatedly be extracted, modified, and re-submitted to a PDM system. If the design and data prove robust, they will end up in a manufacturing environment and lead to an actual, physical product.

The data will be in many formats and varying degrees of completeness throughout this life-cycle. They also contain peripheral information that may be necessary for some processes, such as the mid-plane, or theoretical fillets used by analysis programs. Although the data are not always intentionally captured in a model, they can often be extracted and utilised. In some phases it is acceptable to have only partial data; in other phases, very complete and detailed data are required. In some cases too much data may be provided for the required use.

By understanding the life-cycle of the product data and defining the state of the data and the requirements at each phase and usage of the data, a controlled data flow can be implemented. The requirements, content, and quality of the data can be defined and incorporated at each phase of development, making the correct data available to the correct phase, at the correct time. Immense savings can be realised by understanding this data flow and the requirements at each phase, reducing or eliminating the need for recreating, repairing, or searching for data to be used in each phase of the product development process.

This entire process may begin in an OEM or a supplier, and the data often get passed around and shared in any direction within the supply chain (i.e., up to OEMs, down to lower tiers, or across to counterparts within organisations). When this takes place, the data are transferred by some means (e.g., network, internet, or tape or CD media). They may also be modified by many users and translated into different formats to meet the different requirements of the particular discipline that is using the data at that phase of its development.

If there are changes to the data in later phases of development such as manufacturing, ideally these changes would be communicated back "upstream" to the origin or owner of the design data, or at least to that PDM system. Although this rarely takes place in today's environment, it is crucial and potentially a cause for a great deal of cost. If this modification for manufacturing does not get communicated back upstream, the "master model" in the originating PDM system may have characteristics that cannot be manufactured or features that are not optimised. Therefore, if the design is released in a subsequent model year or design variation, the flaws will still exist. At best, these flaws may get caught by the new manufacturing users. At worst, the flaw could make it through the process this time around; therefore, a different, possibly flawed component may get manufactured.

The product data life-cycle does not end there. The design, characteristics, reliability, effectiveness, warranty costs, and even reusability are often tracked. The value of tracking the product throughout its useful life can affect future designs’ effectiveness, reliability, failure, and ultimately cost. Very often this design will be in use for several "generations" of a product. If the component is a cause of a problem or costs, it should be re-evaluated before being released again.

The product data will eventually be stored for future reference or use and may eventually be archived. There are certain regulatory requirements about keeping and making the data available at some point in the future, often 10 years or more! This can cause great effort and expense in keeping the data and having the appropriate systems available 10 years from now to read the data. The logistics of tracking the data and maintaining these systems and back-up media is a very expensive proposition. In a future release of this document there will be a section on archival data.

2.2 Data Exchange

In today's push for collaboration, supplier integration, and concurrent engineering, data exchange is becoming a more integral part of the product development process. Data exchange may take place any time product data is being shared. Exchanging data can take many forms. The following examples illustrate some scenarios of data exchange.

- CAD A -> CAD A Even exchanging with the same CAD system (native exchange), considerations must be taken in regard to configurations, accuracy, conventions, etc.
- CAD A -> CAD B Typically must be exported from CAD A to some neutral format (IGES, STEP, etc.) and then imported into CAD B from this neutral format or via a direct translator.
- CAD A -> CAX B Where CAX could be CAM, CAE, or another program that uses CAD data. In this instance, the data usually must be translated from the native CAD A file into the CAX B format. Unless a reliable direct translator exists, a neutral format file is likely to be the best approach to exchange.

Because of the necessity of data exchange in the product development process, there have been many studies done in the industry, and one factor always becomes the main factor in limiting successful data exchange data quality. In fact, some studies indicate up to 95 percent of data exchange failures can be traced to poor data quality in the original data.

A data exchange process, methodology, and formal agreement should be considered and implemented in the product development process. Very thorough and robust guidelines regarding this exist in work performed by the VDA, GALIA, and ODETTE (see Attachment C). The VDA recommends that all companies conclude data exchange agreements in accordance with the VDA recommendation 4950 and data quality agreements corresponding to this document (VDA 4955). These should take into consideration individual company-specific rules and standards.

These agreements may have different fields of application. An agreement might apply to only one particular part, whereas another might be applicable to a group of parts within a project. A fundamental agreement at the company level could represent a good beginning and can form the basis for subsequent agreements.

2.3 List of Data Uses and Associated Data Requirements

This chapter presents many of the disciplines that deal with CAD data in some aspect of product development. The users in these disciplines can be creators or consumers of the data. They may only use the data for visualisation, or make very complex mathematical analyses of the data. How users work with the product data determines the formats most useful to them.

Table 3. shows the file formats and geometry types that could be used by the particular discipline listed. This information in the table encompasses how a process could be achieved using a variety of formats. Different companies, organisations, and practices will perform the same task in different ways. This table is meant to present a wide view and capture any data formats that might be used for each aspect of product development. Although there are many other file formats that are not listed here, this table can be used as a general guideline for possibilities of using product data in the disciplines specified.

Table 4. conveys the format that each discipline will possibly use. Of course each organisation will be different, but this table can be used as a starting point to determine which formats a product development organisation prefers. A robust understanding of each of the disciplines' requirements and preferences is crucial to implementing good data flow and data quality. This will not only depend on the product development processes in your organisation, but also will depend on software applications, infrastructure, and process flow, as well as personal techniques, methodologies, and preferences. Note that in **Table** 4. 3D includes all three-dimensional geometry, not just solid models.

Table 3. Data requirements by use

	File Formats / Geometry Types				Geometry type		Other	
Discipline/ Format	Native/ Proprietary	STEP	IGES	STL	2D	3D	Visualisation e.g., VRML	Mesh
CAD								
Design	X	X	X		X	X		
Visualisation	X	X	X	X	X	X	X	
Assembly	X	X		X	X	X	X	
Packaging	X	X		X	X	X	X	
Review	X	X	X	X	X	X	X	
Drawings	X	X	X		X	X	X	
CAM								
Tool Design	X	X	X		X	X		
Simulation	X	X	X			X		X
Assembly	X	X		X		X	X	
Automation	X	X		X		X	X	
Robotics	X	X		X		X	X	
NC	X	X			X	X		
Analysis	X	X				X		X
Deformation	X	X	X			X		X
CAE								
FEA	X	X				X		X
CFD	X	X				X		X
Crash	X	X				X		X
Acoustics	X	X				X		X
Kinematics	X	X				X		X
Rapid Prototype								
STL	X	X	X	X		X		
LOM	X	X	X	X		X		
FDM	X	X	X	X		X		
Inspection								
CMM	X	X	X		X	X		
Gauging	X	X				X		
Fixtures	X	X				X		
PDM								
Config. Mgt.		X					X	
Data Structure		X					X	
Meta-data		X					X	

Table 4. Likely preferred formats

Discipline/ Requirement	Preferences				
	2D	Points	3D	Mesh	Meta-data
CAD					
Design			X		X
Visualisation			X		
Assembly			X		X
Packaging		X	X		X
Review			X		
Drawings	X		X		X
CAM					
Tool Design			X		X
Simulation		X	X		
Assembly			X		X
Automation			X		X
Robotics		X	X		
NC		X	X		X
Analysis			X	X	X
Deformation			X	X	
CAE					
CFD			X	X	X
Crash			X	X	X
Acoustics			X	X	X
Kinematics		X	X	X	X
Rapid Prototype					
STL			X		X
LOM			X		X
FDM			X		X
Inspection					
CMM		X	X		X
Gauging			X		X
Fixtures			X		X
PDM					
Configuration Management					X
Data Structure					X
Meta-data					X

(Blank page)

SECTION II: PDQ CRITERIA

This section describes the specific quality criteria that apply to various kinds of product data. In this version of the guidelines, the content is oriented toward CAD data. Future versions will add criteria for other types of product data, as is shown by the section headings hereafter.

PDQ Criteria Coding System

This set of guidelines covers many different types of product data. For clarity and to support the potential use of the content in formats other than a printed document, a coding system has been established that explicitly identifies each of the checking criteria. This system is used throughout the guidelines.

The coding system uses the structure shown here.

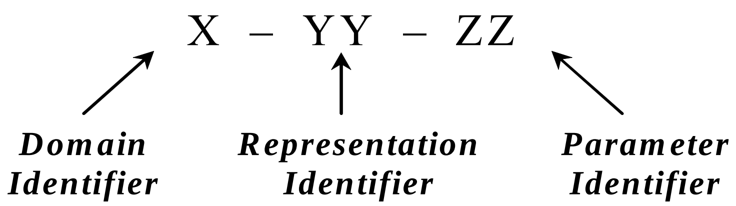


Figure 3. Structure of criteria code

An explanation of the codes is provided in the following list.

Criteria Code structure values.

Values allowed for the **Domain Identifier**:

A	**A**nalysis	**M**	**M**anufacturing
D	**D**rawing	**N**	**N**umerical control
G	**G**eometry - CAD data	**O**	N**O**n-geometric
I	**I**nspection	**P**	**P**roduct data management

Values allowed for the **Representation Identifier:**

AR	**A**ssembly **R**epresentation	**OR**	**O**the**R**
CM	**C**AD **M**odel	**PE**	**PE**ntahedron
CS	**C**o-ordinate **S**ystem	**PR**	**PR**esentation
CU	**CU**rve	**PY**	**PY**ramid
ED	**ED**ge	**QU**	**QU**adrilateral
EL	**EL**ements	**SH**	**SH**ell

FA	**Fa**ce	**SK**	**SK**etch
FE	**Fe**atures	**SO**	**SO**lid representation
GE	**Ge**ometry	**SU**	**SU**rface representation
GL	**G**roup / **L**ayer	**TE**	**TE**trahedron
HE	**HE**xahedron	**TR**	**TR**iangle
LO	Edge **Lo**op		

Values allowed for the **Parameter Identifier**:

AD	**A**ssociative **D**imension	**LU**	**L**ayer **U**sage
AN	**AN**alytical (previously non-nurbs)	**LW**	**L**ine **W**idth
AP	**A**ccuracy **P**arameter	**LY**	**L**a**y**er Used
AR	**A**ssembly **R**elationship	**MA**	**M**inimum **A**ngle
AS	**AS**pect ratio	**MH**	**M**issing **H**istory
CL	**CL**osed element	**MU**	**MU**ltiple elements
CN	**C**o-ordinate system **N**ame	**NA**	**NA**rrow element
CO	**CO**lour settings	**NC**	**N**ot fully **C**onstrained sketch
	COntinuity	**ND**	**N**umber of **D**rawing sheets
CR	**C**urvature **R**adius	**NG**	**N**umber of **G**roups
CS	More than one **C**o-ordinate **S**ystem	**NL**	**N**umber of **L**ayers
CV	**C**AD **V**ersion	**NM**	**N**on-**M**anifold
DC	**D**egenerate **C**urve	**NO**	co-ordinate system orientation
DI	**DI**mensions	**NR**	**N**on-**R**eference co-ordinate system active
DL	2D/3**D** **L**inkage	**NS**	**N**on-**S**mooth curvature between elements **-** (G_2 discontinuity)
DM	**D**isplay **M**ode	**NT**	**N**on-**T**angent angle between elements **-** (G_1 discontinuity)
DP	**D**egenerate at **P**oint	**OB**	**O**utside bounding **B**ox
DR	Number of **DR**awing sheets	**OU**	**O**ver-**U**sed element
DU	2D **D**rawing not **U**pdated	**PA**	Middle **P**oint **A**lignment
EC	**E**lement **C**olour	**PD**	Middle **P**oint **D**eviation
ED	**E**lement identifier **D**isplay	**PE**	**P**rohibited **E**lement
EE	**E**ncapsulated **E**ntities	**PF**	**P**lot **F**rame points
EG	**E**dge **G**ap	**PN**	**P**hysical file **N**ame
EI	**E**xternal **I**tem reference	**PT**	**P**oin**t** marker Symbol
EL	**E**mpty **L**ayer group	**RN**	**R**elatively **N**arrow
EM	**EM**bedded elements	**RS**	**R**eference **S**et
EN	**E**lement **N**ame	**SA**	**S**harp **A**ngle
EP	**E**mpty encapsulated entities **P**resent	**SC**	**S**pecial **C**haracter

ER	**E**xplicit **R**eference		ISO **C**onformable text
	External database **R**eference	**SE**	cad **S**tartup **E**nvironment
EV	**E**mpty drawing **V**iew	**SK**	**SK**ew angle
FD	**F**ake **D**imensions	**SM**	**S**ize of **M**odel
FG	**F**ra**G**mented	**SN**	cad **S**ource **N**otice
FO	**FO**lded element	**SP**	**S**implified **P**art
FR	**FR**ee element	**SR**	**S**creen **R**efit
	FRee faces	**SS**	**S**cale
FS	**F**ile **S**ize	**ST**	**ST**retch
GL	**L**ayer **G**roup	**SU**	**U**nit
GN	**G**roup **N**ame	**TA**	**TA**per
GU	**G**roup **U**sed	**TI**	**TI**ny elements
HD	**H**igh **D**egree	**TS**	**T**ransformation **S**tored
HN	**H**istory **n**ot Used	**UC**	**U**ndefined assembly **C**onstraints
HU	**H**istory not **U**pdated	**UD**	**U**ser-**D**efined element
HY	**HY**brid	**UE**	**U**nused **E**lement
IC	**I**tem data **C**onsistency	**UF**	**U**nresolved **F**eature
ID	**I**nappropriate **D**egree	**UH**	**U**nused **H**istory
IE	**I**dentical encapsulated **E**ntity	**UN**	**UN**used elements
IF	**I**nactive **F**eature	**UP**	**U**nused enca**p**sulated entities
IG	**I**dentical element within many **g**roups	**VD**	**V**iew **D**ependent object
IK	**I**ndistinct **K**nots	**VE**	**E**lement **V**isibility
IN	**I**tem **N**ame	**VF**	**V**iew **F**rames
IP	**I**tem **P**roperty	**VG**	**V**ertex **G**ap
IR	**I**tem **R**eference	**VN**	**V**iew **N**ame
IS	**I**nter**S**ection	**VO**	**Vo**id
IT	**I**nconsistent **T**opology	**VP**	**V**iew **P**rojection
JA	**JA**cobian	**WA**	**WA**rpness
LA	**LA**yer group	**WD**	**W**rong **D**egree
LG	**L**arge **G**ap between elements **-** (G_0 discontinuity)	**WL**	**W**rong **L**ayer distribution of instances
LN	**L**ayer **N**ame	**WV**	**W**a**V**y element
LS	**L**ocal co-ordinate **S**ystem	**XD**	E**X**ternal 2D **D**rawing
LT	**L**ine **T**ype		

3 CAD Data

This chapter describes the product data quality criteria associated with CAD data. It is divided into three sub-chapters:

- 3.1 Geometric Quality Criteria Descriptions
- 3.2 Non-Geometric Quality Criteria Descriptions
- 3.3 Drawing Quality Criteria Descriptions

Note: All references to distance in the CAD data criteria in this section imply geometric distances unless otherwise specified.

3.1 Geometric Quality Criteria Descriptions

Geometrical data quality provides information about how, and with what precision, geometry elements shall be generated, so that the subsequent usability of these elements within the process chain is possible. This section is organised along the natural hierarchy of data, beginning with curves and ending with models. In all cases, the criterion addressing more complex geometry assumes that the underlying geometry also satisfies the relevant criteria. For example, when applying the solid model criteria, the model is expected to also meet the criteria for curves, surfaces, edges, edge loops, faces, and shells.

Table 5 shows the relationships of the geometric quality criteria contained in this section. The rows of the matrix represent categories of problems. The columns of the matrix represent categories of geometric entities. Each cell of the matrix contains the title of the quality criterion (if there is one) that addresses that row's category of problem in the context of that column's geometric entity. Combining the encoding contained in the column heading with that of the row heading gives the encoding for the specific criterion. Thus, "Inconsistent face on surface" is the criterion that addresses inconsistent topology in faces and has the encoding G-FA-IT (3.1.5.5 Inconsistent face on surface: G-FA-IT). The following subsections also correspond to the matrix columns.

Table 5. Geometric data criteria (number and encoding terms, by row and column) Page 1 of 3

		ENTITY CATEGORY and CRITERIA NUMBER						
CATEGORY	Quality	Curve	Surface	Edge	Edge Loop	Face	Shell	Solid
NAME	CODE	G-CU	G-SU	G-ED	G-LO	G-FA	G-SH	G-SO
G0 Discontinuity	LG	Large segment gap 3.1.1.1	Large patch gap 3.1.2.1		Large edge gap 3.1.4.1		Large face gap 3.1.6.1	
G1 Discontinuity	NT	Non-tangent segments 3.1.1.2	Non-tangent patches 3.1.2.2				Non-tangent faces 3.1.6.2	
G2 Discontinuity	NS	Non-smooth segments 3.1.1.3	Non-smooth patches 3.1.2.3				Non-smooth faces 3.1.6.3	
Edge Gap	EG					Large edge face gap 3.1.5.1		
Vertex Gap	VG					Large vertex gap 3.1.5.2		
Small Curvature Radius	CR	Curve with a small radius of curvature 3.1.1.9	Surface with a small radius of curvature 3.1.2.14					
Wavy	WV	Wavy planar curve 3.1.1.11	Wavy surface 3.1.2.16					
Folded	FO		Folded surface 3.1.2.18					
Degenerate Curve	DC		Degenerate surface boundary 3.1.2.4					
Degenerate at Point	DP		Degenerate surface corner 3.1.2.5					
Sharp Angle	SA				Sharp edge angle 3.1.4.4		Sharp face angle 3.1.6.9	

Table 5. Geometric data criteria (number and encoding terms, by row and column) Page 2 of 3

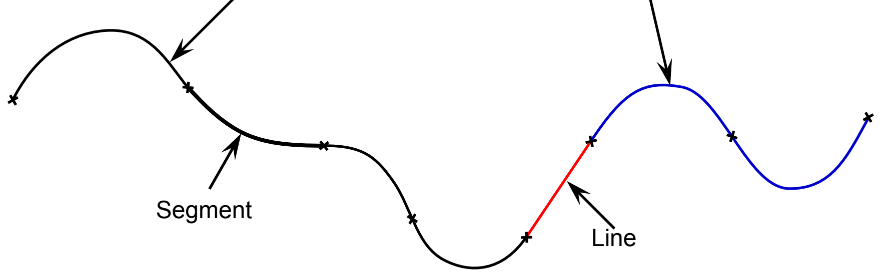
CATEGORY NAME	Quality CODE	Curve	Surface	Edge	Edge Loop	Face	Shell	Solid
		G-CU	G-SU	G-ED	G-LO	G-FA	G-SH	G-SO
Tiny	TI	Tiny curve or segment 3.1.1.10	Tiny surface or patch 3.1.2.12	Tiny edge 3.1.3.5		Tiny face 3.1.5.10		Tiny solid 3.1.7.4
Narrow	NA		Narrow surface or patch 3.1.2.10			Narrow face 3.1.5.8		
Relatively Narrow	RN		Relatively narrow neighboring patches 3.1.2.11			Narrow Region 3.1.5.9		
Intersection	IS	Self-intersecting curve 3.1.1.6	Self-intersecting surface 3.1.2.8		Self-intersecting loop 3.1.4.3	Intersecting loops 3.1.5.6	Self-intersecting shell 3.1.6.6	Intersecting shells 3.1.7.1
Analytical	AN			Analytical edge 3.1.3.1		Analytical face 3.1.5.3		
Indistinct Knots	IK	Indistinct curve knots 3.1.1.5	Indistinct surface knots 3.1.2.7					
Inappropriate Degree	ID	Inappropriate degree linear curve 3.1.1.12	Inappropriate degree planar surface 3.1.2.19					
High-Degree	HD	High-degree curve 3.1.1.4	High-degree surface 3.1.2.6					
Fragmented	FG	Fragmented curve 3.1.1.7	Fragmented surface 3.1.2.9	Fragmented edge 3.1.3.4				
Closed	CL			Closed edge 3.1.3.2		Closed face 3.1.5.4		
Inconsistent Topology	IT			Inconsistent edge on curve 3.1.3.3	Inconsistent edge in loop 3.1.4.2	Inconsistent face on surface 3.1.5.5	Inconsistent face in shell 3.1.6.5	

Table 5. Geometric data criteria (number and encoding terms, by row and column) Page 3 of 3

CATEGORY	Quality	Curve	Surface	Edge	Edge Loop	Face	Shell	Solid
NAME	CODE	G-CU	G-SU	G-ED	G-LO	G-FA	G-SH	G-SO
Free	FR						Free Edge 3.1.6.4	
Non-Manifold	NM						Over used edge 3.1.6.7	
Over-used	OU						Over used Vertex 3.1.6.8	
Multiple	MU		Multi-face surface 3.1.2.17					Multi-volume solid 3.1.7.2
Embedded	EM	Embedded curves 3.1.1.8	Embedded surfaces 3.1.2.13			Embedded Faces 3.1.5.11		Embedded solids 3.1.7.3
Unused	UN		Unused patches 3.1.2.15					
Void	VO							Solid void 3.1.7.5

3.1.1 Curve

Points, curves and lines are regarded as part of the wire geometry. They serve, for example, as a geometry aid for the generation of faces and solids, as contours for NC programming, as well as in drawings.



Example: Curve

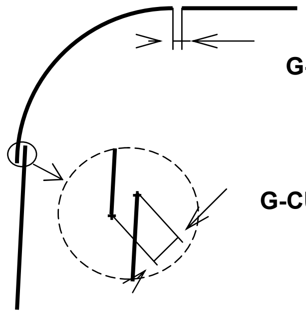
3.1.1.1 Large segment gap (G_0 discontinuity): G-CU-LG

Problem description: Large distance between or overlapping of adjacent curve segments - a G_0 discontinuity.

Measurement: Distance between segment endpoints at common bound.

Supporting information: The first and most important continuity issue is “position continuity,” i.e., the transition of curves and curve segments without gaps and/or overlapping end points. A position discontinuity endangers follow-up operations that build upon the unity of curve paths, especially after scaling, offsetting, or transfer.

Recommendation: Position discontinuities are to be rectified by limiting the affected curves to one another within the tolerances. A possible, necessary extension or trim of one or both elements is preferred.



Example: Large segment gap (G_0 discontinuity)

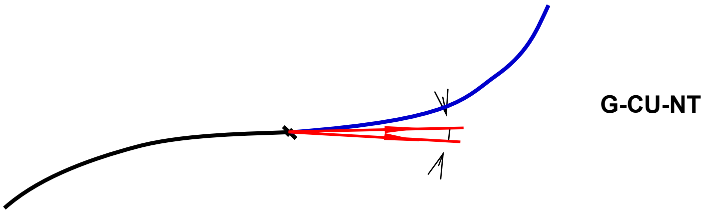
3.1.1.2 Non-tangent segments (G_1 discontinuity): G-CU-NT

Problem description: Non-tangent angle between adjacent curve segments—a G_1 discontinuity.

Measurement: Angle between segment tangent vectors at common bound.

Supporting information: Tangential continuity (with specified position continuity) means the kink-free transition of two curves without a change in the tangential angle. A tangential discontinuity is generally visible and can be felt. In a fully rounded curve, this generally occurs unintentionally. There can also be intentional design necessitated tangential discontinuities (e.g., chamfers/bevels, character lines).

Recommendation: Interactively correct the curves by recreating them with identical tangential conditions, or round off with an additional curve with suitable tangential specifications (i.e., round off two straight with a radius).



Example: Non-tangent segments (G_1 discontinuity)

3.1.1.3 Non-smooth segments (G_2 discontinuity): G-CU-NS

Problem description: Large curvature change between adjacent curve segments, a G_2 discontinuity.

Measurement: Curvature continuity at the contact point of two segments (by a given position/tangential continuity) means:

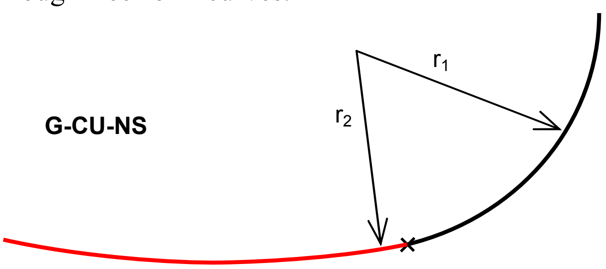
a) Central points of curvature radii lie on same side of planar segments (not relevant for 3D segments)

b) Difference of absolute values of radii, divided by mean value of radii, is below the given accuracy, that is:

$$G-CU-NS=\frac{2|r_1-r_2|}{|r_1|+|r_2|}$$ (Note: G-CU-NS is always positive)

Supporting information: Curvature continuity of curves is normally only required by the contour description of component parts with special functions (cams, worms, etc.), or by stylistic elements.

Recommendation: Replace the affected elements by elements with suitable curvature conditions at the ends; e.g., neighbouring elements, which in each case have constant curvatures (straight lines, circles, etc.), shall be replaced through free-form curves.



Example: Non-smooth segments (G_2 discontinuity)

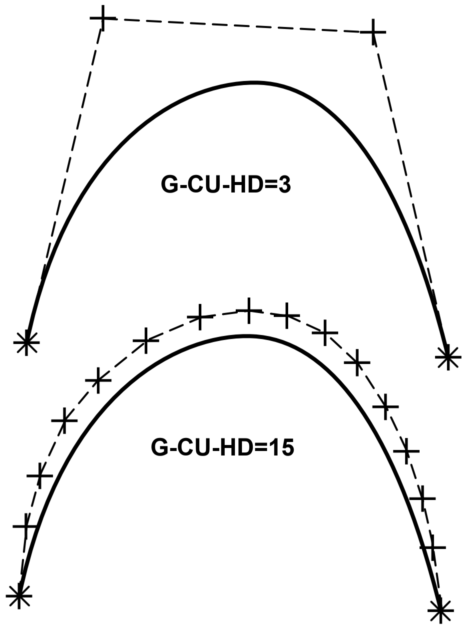
3.1.1.4 High-degree curve: G-CU-HD

Problem description: Degree of polynomial curve is too high.

Measurement: Degree of polynomial curve.

Supporting information: The degree of the polynomial depiction for a curve segment determines the number of degrees of freedom (variance) of a curve. The higher the degree, the greater the complexity of the curve. Curves with high polynomial degrees are susceptible for unintentional or unwanted definition and curvature and therefore, where appropriate, must be approximated when translated to another CAD, CAM, or CAE system, i.e., approximated within the bounds of the given accuracy. In both cases, this generally means a worsening of data quality.

Recommendation: High-degree polynomial curves should be avoided. High-degree polynomial curves may be subdivided into curves of smaller degree with respect to the given accuracy.



Example: High-degree curve

3.1.1.5 Indistinct curve knots: G-CU-IK

Problem description: Curve has consecutive, non-multiple knot values with real values that are too close to each other.

Measurement: Minimum, non-zero difference between consecutive knot values.

Supporting information: A knot vector is required for the definition of NURBS and B-Spline curves. This defines, among other things, the number of the curve segments and the continuity of the transitions between the individual curve segments. The knot vectors are defined through a series of real numbers. Individual knots can be positioned on top of one another. This is called "Multiple-weighting of knots" or, in short, "Multiple knots." Curves with close neighbouring knots can be changed in their internal continuity characteristics. This can happen through knots coinciding with one another during the transfer into another system environment set with coarser tolerances.

Example of a knot vector of a NURBS-curve of three degrees: (0.0, 0.0, 0.0, 0.0, 0.3333, 0.3334, 1.0, 1.0, 1.0, 1.0)

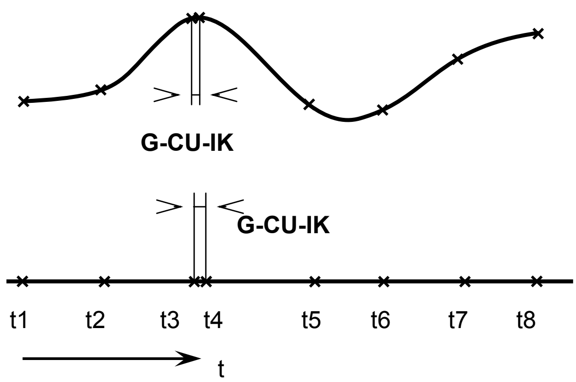
Knot accuracy < 0.0001: - Curve consists of three curve segments,

- Internal segment transitions are C2 continuous.

Knot accuracy > 0.0001: - Curve consists of two curve segments,

- Internal segment transition is C1 continuous.

Recommendation: Regenerate curves with sufficiently large knot clearances.



Example: Indistinct curve knots

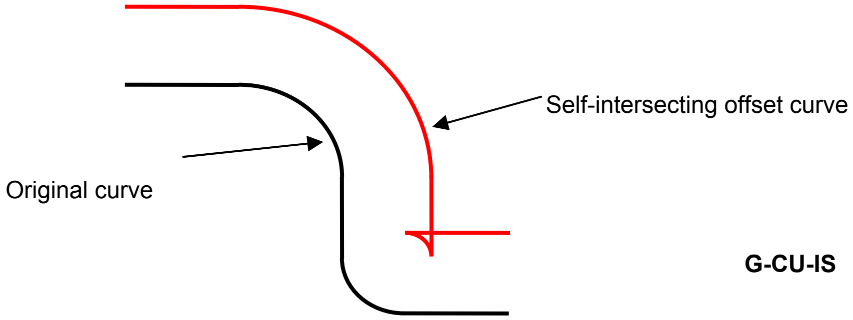
3.1.1.6 Self-intersecting curve : G-CU-IS

Problem description: Curve intersects itself at one or more locations that are not both endpoints.

Measurement: Whether a curve intersects itself within the designated (system or otherwise) accuracy.

Supporting information: A self-penetration/intersection is the existence of an intersecting point of a curve with itself. It is always unintentional, having no design purpose. This error causes problems with other geometrical operations, such as the generation of offsets or faces, as well as with NC programming.

Recommendation: Self-penetration often results from faulty development of offsets (offset distance is larger than the inside radius) or projections (three-dimensional curves in one plane) and are to be avoided wherever possible. Retroactively regenerate the curves correctly.



Example: Self-intersecting curve

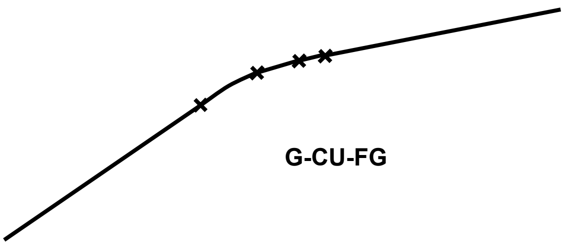
3.1.1.7 Fragmented curve: G-CU-FG

Problem description: Curve is defined by too many segments.

Measurement: Count of segments in curve.

Supporting information: An unreasonably high number of segments within a curve is generally a sign of unfavourable complexity of a curve. This occurs, for example, through merging of different curves or a poor approximation of a curve of higher degree to that of lower degree.

Recommendation: Replace the curve with another curve with as few segments as possible. A curve with harmonic curvature distribution and a large number of (smaller) segments can be replaced where necessary through curves with meaningful, higher degrees. (Re-computation may be necessary under observation of the given accuracy.)



Example: Fragmented curve

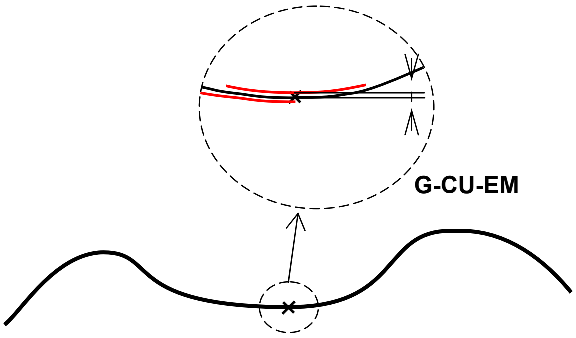
3.1.1.8 Embedded curves: G-CU-EM

Problem description: Set of curves with one that completely overlaps the other(s). Set can include curves of any type.

Measurement: Whether there is a curve completely embedded within another curve within the designated accuracy.

Supporting information: By miscellaneous geometrical operations, or through copying external geometry into the model, (approximately) identical elements can occur that unnecessarily enlarge the space requirements of the model and cancel out the validity of the original element. For example, identical elements, also known as double elements, often impede the automatic recognition of continuous curved lines or impede NC and FEA operations. Also, elements that lay completely in a larger element are considered to be identical.

Recommendation: Delete one of the double elements. It is important to take care as to which of the double elements shall be deleted; consider usage and parent/child relationships.



Example: Embedded curves

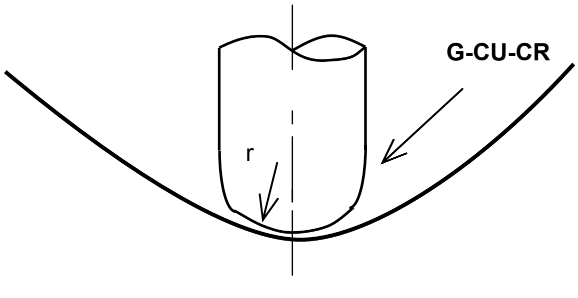
3.1.1.9 Curve with a small radius of curvature: G-CU-CR

Problem description: Curve has a small radius of curvature value.

Measurement: Minimum radius of curvature along curve.

Supporting information: Such curves may cause problems in offset curve creation or may create degenerate surfaces and cause problems in downstream use such as finite element mesh generation and machining. For example, to guarantee the ability to machine along a curve, the radius of curvature must not fall short of the given minimum at any position; otherwise, lesions can occur on that curve during the work process.

Recommendation: Curves with curvature less than the given minimum must be recreated, e.g., through approximation or smoothing.



Example: Curve with a small radius of curvature

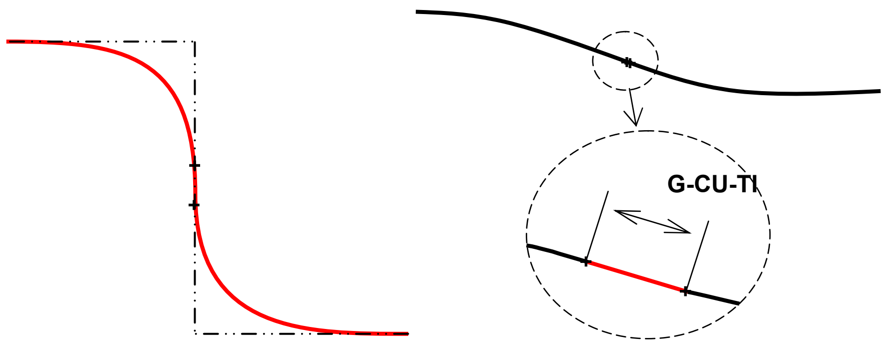
3.1.1.10 Tiny curve or segment: G-CU-TI

Problem description: Overall extent of curve or segment is too small.

Measurement: Length of curve or segment.

Supporting information: Elements that fall short of a particular size by particular geometrical operations (i.e., scaling, generation of offsets), by the exchange of data (in a system of lesser accuracy), or through further processing (NC) can lead to invalid elements and thereby to gaps. Reworking these elements means a considerable increase in effort. These elements often occur involuntarily, not only through filleting but also through "closing mechanisms" during bridging of small gaps or by overlapping features or entities.

Recommendation: Eliminate tiny elements through an appropriate extension (extrapolation) of the elements to be joined and delete the corresponding small elements or segments. Alternatively, enlarge the tiny elements and join the corresponding element.



Example: Tiny curve or segment

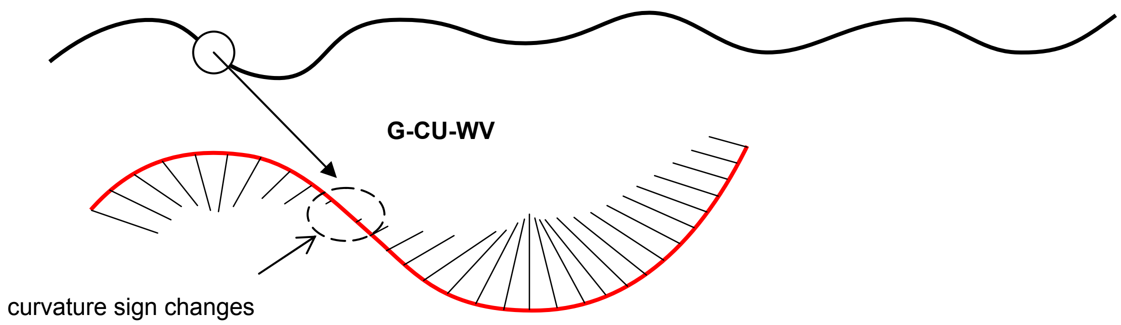
3.1.1.11 Wavy planar curve: G-CU-WV

Problem description: Curve has too many curvature sign changes.

Measurement: Count of curvature sign changes.

Supporting information: A waviness, i.e., a number of algebraic sign changes on the curvature of a free form planar curve, is often unintentional and perhaps critical for following operations, e.g., by the generation of an offset. This problem occurs in three-dimensional curves as well.

Recommendation: Analyse the tangential and restart point conditions of the curve and clean up or, where necessary, recreate. Also, analyse the created faces of intersecting curves and correct or reconstruct where necessary.



Example: Wavy planar curve

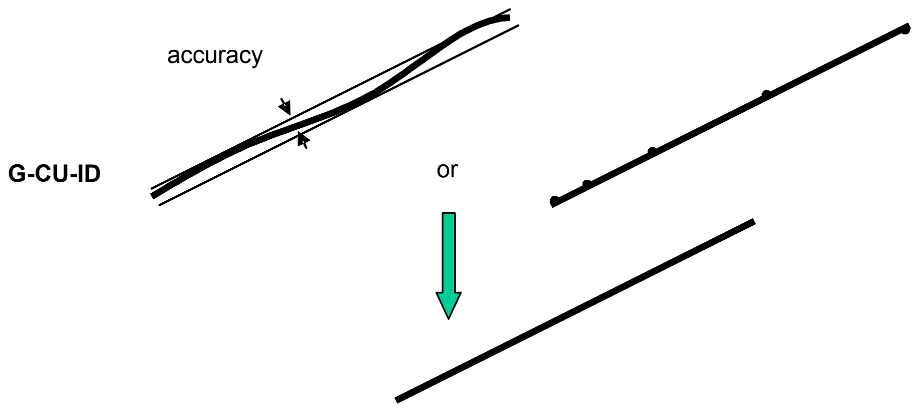
3.1.1.12 Inappropriate degree linear curve: G-CU-ID

Problem description: A linear or nearly-linear curve is defined with too high a degree.

Measurement: Degree of nearly-linear curve, within a given accuracy, or degree of a straight curve.

Supporting information: Over-specified polynomial curves can cause multiple errors when used by adjoining elements in the model or when translated to other systems. When translated, the result might be a curve with local areas of high curvature or even self-intersection.

Recommendation: Replace the polynomial curve with a line or reduce the degree of the curve down to 1.

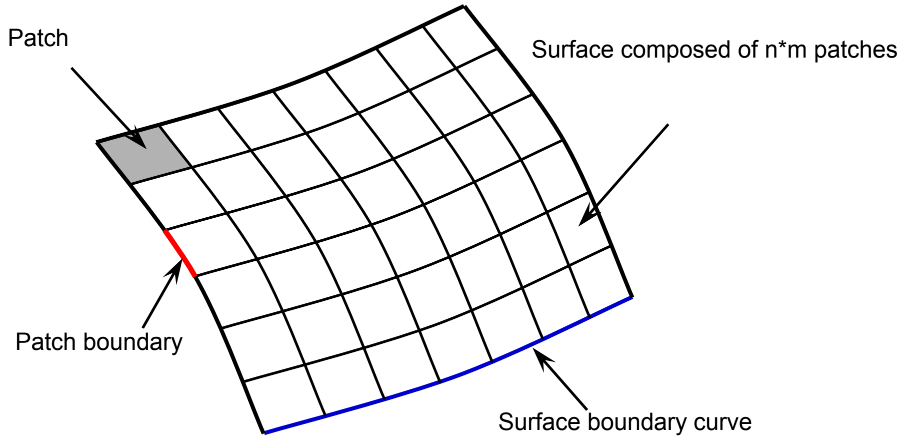


Example: Inappropriate degree linear curve

3.1.2 Surface

Surface is the name given to the basic mathematical representation of a geometric surface element bounded by surface boundary curves. The surface extent of a part may protrude beyond its actual contours.

Surfaces may be composed of several surface segments called "patches." These can be bound together within the limits of internal tolerances for position and gradients. A surface is formed from a group of *n* times *m* patches (where *n* and *m* are the number of segments of the surface boundary curves).



Example: Surface

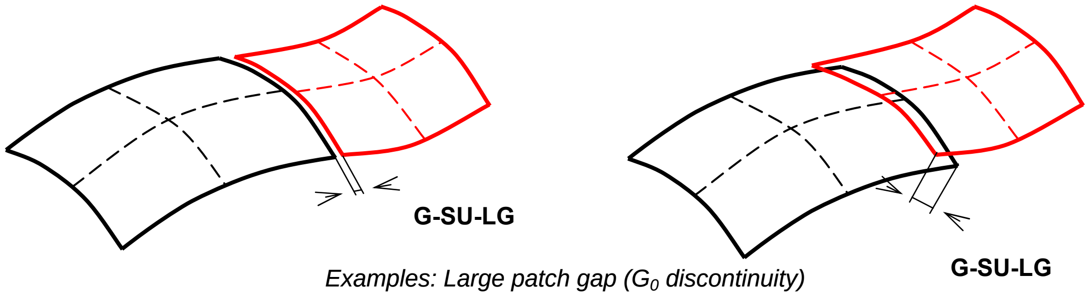
3.1.2.1 Large patch gap (G_0 discontinuity): G-SU-LG

Problem description: Large distance between or overlapping of adjacent surface patches—a G_0 discontinuity.

Measurement: Maximum distance between pairs of nearest points along proximate boundaries.

Supporting information: Similar to the continuity of curves, the position, tangential, and curvature continuities of surfaces are of considerable importance for their qualities as basic geometry (i.e., for bounded surfaces or intersection curves). This condition occurs most often when adjacent patches are generated using different curves. In this criterion, only the continuity within the surface segments (patches) is considered.

Recommendation: Naturally bounded surfaces with discontinuity of the patches must be corrected or regenerated via suitable fundamental conditions.



Examples: Large patch gap (G_0 discontinuity)

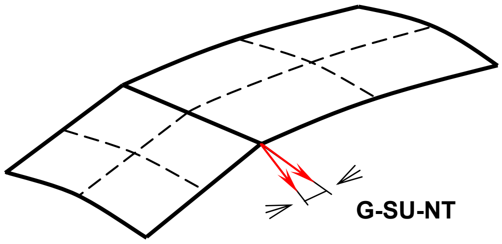
3.1.2.2 Non-tangent patches (G_1 discontinuity): G-SU-NT

Problem description: Non-tangent angle between adjacent surface patches—a G_1 discontinuity.

Measurement: Maximum angle between patch normals evaluated at adjacent points along coincident boundaries (provided that G_0 continuity is given).

Supporting information: Tangential continuity (given position continuity) means the kink-free transition of two adjacent surface patches without a change in the tangential angle below given accuracy. A tangential discontinuity may be visible or felt. In practical use, the acceptable angle difference is related to the neighbouring patch size (larger angles tend to be acceptable with smaller patches).

Recommendation: Interactively correct the surface by recreating or modifying it with tangential conditions.



Example: Non-tangent patches (G1 discontinuity)
Normal angles shown for visualisation

3.1.2.3 Non-smooth patches (G_2 discontinuity): G-SU-NS

Problem description: Large curvature change between adjacent surface patches—a G_2 discontinuity.

Measurement: Curvature continuity at the contact point of two patches (by a given position/tangential continuity) means:

a) Check Curvature continuity in consecutive normal section planes.

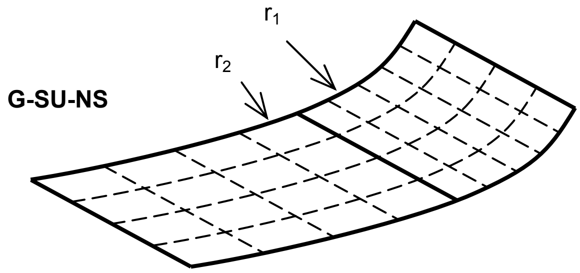
b) Central points of curvature radii lie on same side of the patches.

c) Difference of absolute values of radii, divided by mean value of radii, is below the given accuracy,that is:

$$G-SU-NS=\frac{2|r_1-r_2|}{|r_1|+|r_2|}$$ (note: G-SU-NS is always positive)

Supporting information: Curvature continuity of surfaces is normally required only by the description of component parts with special functions (cams, worms, etc.) or by stylistic elements.

Recommendation: Replace the affected elements by elements with suitable curvature conditions at the common boundary; e.g., neighbouring elements that have constant curvatures (cylinder, sphere, planar elements, etc.) shall be replaced through free-form surfaces.



Example: Non-smooth patches (G_2 discontinuity)

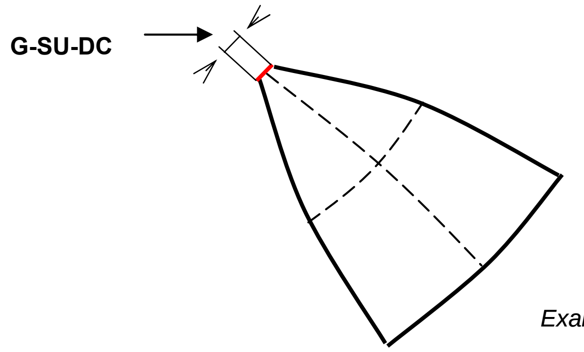
3.1.2.4 Degenerate surface boundary: G-SU-DC

Problem description: Surface or patch has one boundary that is too short. Note that some systems define a valid triangular surface or patch by setting one boundary length below a specified limit.

Measurement: Length of degenerate surface or patch boundary, with this length below the given accuracy for "Tiny" but greater than the chosen accuracy for "zero length."

Supporting information: A surface segment (patch) with exactly one boundary below accuracy ("quasi-triangular patch") can lead to non-defined normal vector, thereby affecting the usability of the patch (e.g., patch offset).

Recommendation: Manually adjust for segment edge sizes larger than the tiny element tolerances or describe by means of a three-sided bounded face (note that some systems do not support triangular patches at all). This solution means create a larger rectangular underlying surface and trim back to the desired three-sided face.



Example : Degenerate surface boundary

3.1.2.5 Degenerate surface corner: G-SU-DP

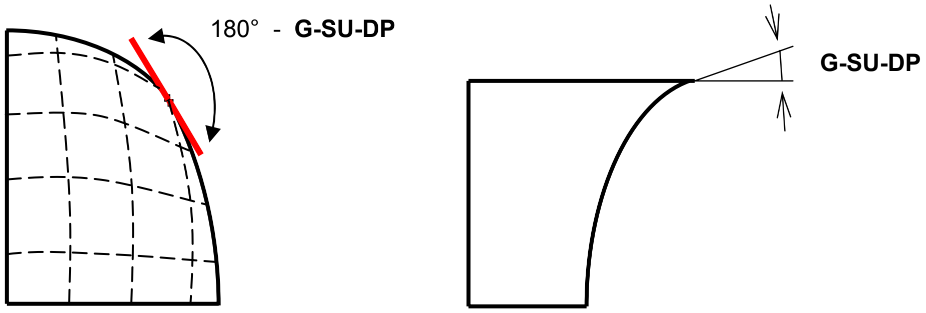
Problem description: Surface corner forms a sharp or tangent angle.

Measurement: Angle between tangents of geometrically adjacent surface boundaries at a surface corner.

Supporting information: If the angle between two neighbouring boundary curves of a surface is less than the minimum angle or more than the maximum angle, this can result in undefined or undesirable surface normal in the corner points.

Recommendation: Divide the surface (e.g., star-shaped from the centre of the surface into three surfaces) or enlarge the surface and generate the required area as a face.

If, despite a critical angle, the normals in the surface corners are well-defined, then these cases may possibly be acceptable.



Example: Degenerate surface corner

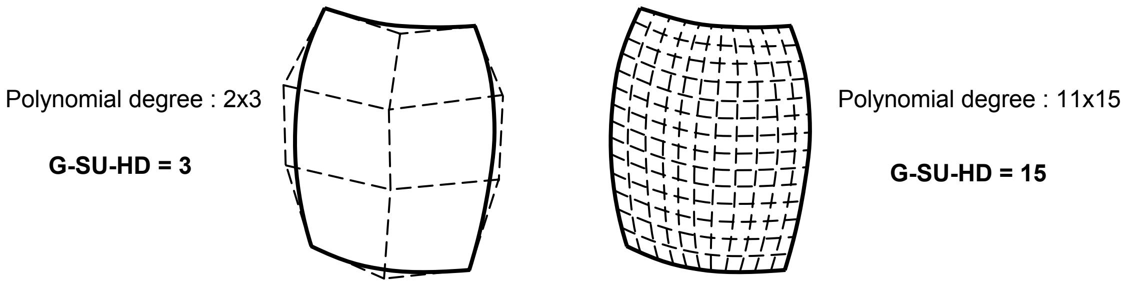
3.1.2.6 High-degree surface: G-SU-HD

Problem description: Degree of polynomial surface is too high.

Measurement: Degree of polynomial surface.

Supporting information: The polynomial degree of the mathematical representation for every patch determines the modelling degrees of freedom of the surface. A too-high polynomial degree can lead to oscillations or, in the case of a reduction in the degree through approximation, to a deterioration of the data quality with respect to the integrity of form. The size of the model can also be increased by overly defined surfaces.

Recommendation: Avoid high polynomial degrees wherever possible. Avoid unnecessarily complex surfaces, or divide them into individual surfaces with smaller degrees dependent upon curvature.



Example: High-degree surface

3.1.2.7 Indistinct surface knots: G-SU-IK

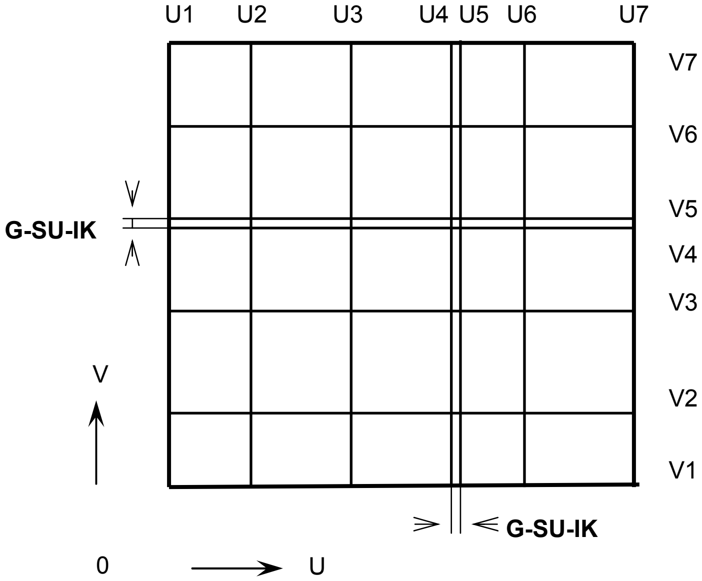
Problem description: Surface has consecutive, non-multiple knot values with real values that are too close to each other.

Measurement: Minimum, non-zero difference between consecutive knot values.

Supporting information: As is the case for NURBS and B-Spline curves, a knot vector for every parameter direction will be required for the definition of NURBS and B-Spline surfaces. These define the number of patches in the parameter directions *u* and *v* and the continuity of the transitions between them. The knot vector will be defined through a series of real numbers. Individual knots can also be identical; these are known as “multiple weighting of knots” or simply” multiple knots.”

After being transferred into another system environment with coarser tolerances, close neighbouring knots can possibly be treated as identical there and consequently the internal continuity within the surface can be changed in an undesirable manner.

Recommendation: Regenerate surface using a more evenly spaced knot definition, or delete and recreate surface if appropriate.



Example: Indistinct surface knots

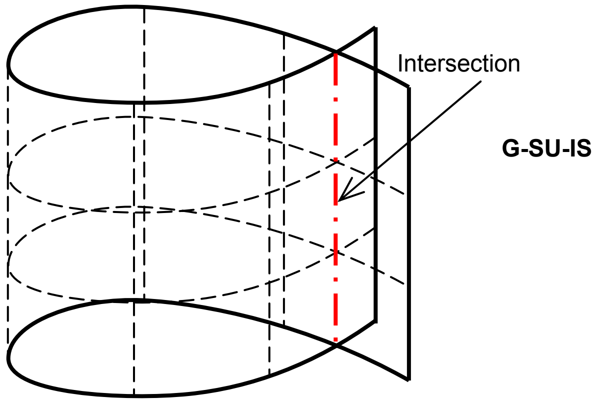
3.1.2.8 Self-intersecting surface: G-SU-IS

Problem description: Surface or surface patch intersects itself.

Measurement: Whether or not the surface or patch intersects itself within the designated (system or otherwise) accuracy.

Supporting information: A self-intersection is the existence of a single curve in two different parametric locations on a surface. It is always unintentional having no design purpose. This error causes problems with solids (leading to self-intersecting faces) and with other geometrical operations, such as the generation of offsets or faces, as well as with downstream data uses such as finite element analysis and NC programming. Self-intersection often results from faulty development of offsets (offset distance is larger than the inside radius) and are to be avoided wherever possible.

Recommendation: Retroactively regenerate the surfaces correctly.



Example: Self-intersecting surface

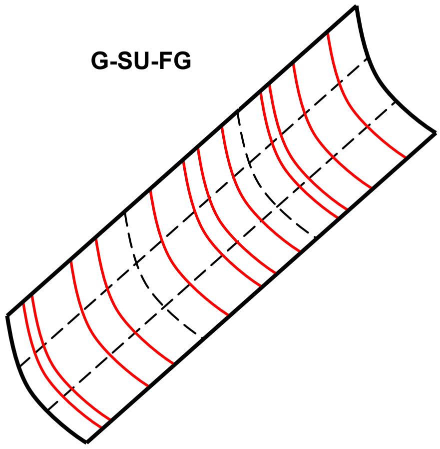
3.1.2.9 Fragmented surface: G-SU-FG

Problem description: Surface is defined by too many patches.

Measurement: Count of patches in surface.

Supporting information: An unreasonably high number of patches within a surface is generally a sign of unfavourable complexity of a surface. This occurs, for example, through a poor approximation of a surface or its creating curves of higher degree to that of lower degree, or through the amalgamation of elements with completely different segmentation.

Recommendation: Partition surfaces with large curvature differences. A surface with harmonic curvature distribution and a large number of (smaller) segments may be approximated where necessary through surfaces with meaningful, higher degrees.



Example: Fragmented surface

3.1.2.10 Narrow surface or patch: G-SU-NA

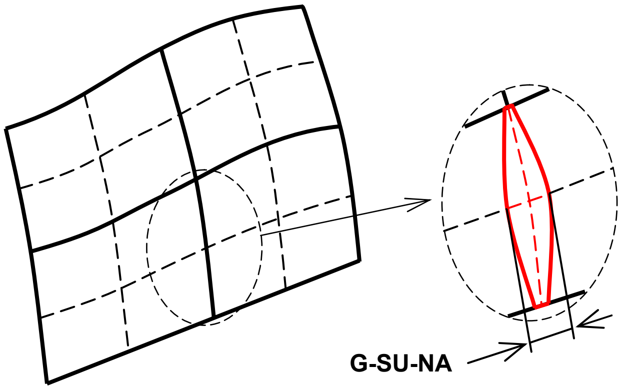
Problem description: Surface or patch is too narrow in either direction compared to recommended minimum value.

Measurement: Maximum distance (in a parametric direction) between patch boundaries.

Supporting information: Very small surfaces or patches can cause considerable problems for further geometry creation and downstream applications. Patches that fall short of a particular extent in at least one direction can result in defective elements. Changes in the system or in the accuracy range can cause this problem and can lead to gaps in the topology. Reworking these elements requires considerable effort.

In addition, narrow elements raise the storage requirements (file size), increase the effort required to make changes, and raise the dangers of continuity problems. They often occur through system automation without the user's knowledge or intent. The automatic closure of gaps in the case of data importation from foreign systems also causes these types of flaws.

Recommendation: Narrow patches should be avoided or eliminated through suitable enlargement and division of the neighbouring elements and then subsequently deleted.



Example: Narrow surface or patch

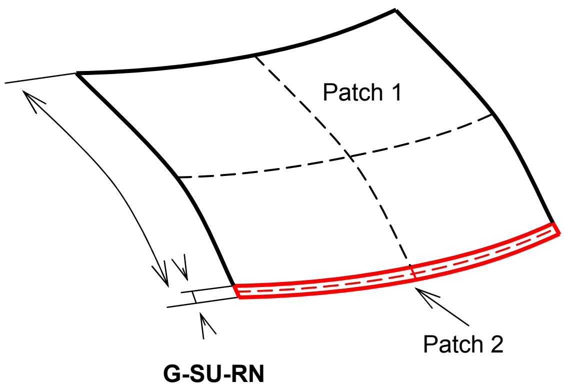
3.1.2.11 Relatively narrow neighbouring patches: G-SU-RN

Problem description: Patch is too narrow compared to its neighbouring patches.

Measurement: Ratio of the linear patch sizes of two adjacent patches in all possible parametric direction combinations.

Supporting information: One patch should not be significantly narrower than neighbouring patches. Such size ratios are a sign of poor partitioning. They may cause problems in mesh generation and surface modification.

Recommendation: Narrow patches should be avoided or made redundant through suitable enlargement and division of the neighbouring elements and then subsequently deleted.



Example: Relatively narrow neighbouring patches

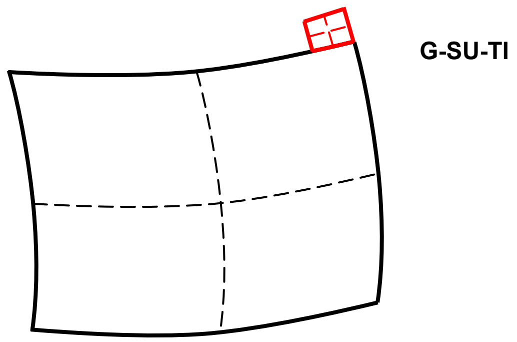
3.1.2.12 Tiny surface or patch: G-SU-TI

Problem description: Overall extent of surface or patch is too small.

Measurement: Area of surface or patch.

Supporting information: Elements that fall short of a particular size can lead to invalid elements and thereby to gaps. This can occur from particular geometrical operations (i.e., scaling, generation of offsets), by the exchange of data (in a system of lesser accuracy), or through further processing (finite element analysis, NC, etc.). Reworking these elements means a considerable increase in effort.

Recommendation: Eliminate tiny elements through an appropriate extension (extrapolation) of the elements to be joined and delete the corresponding small surfaces or patches. Alternatively, enlarge the tiny elements and join the corresponding element.



Example: Tiny surface or patch

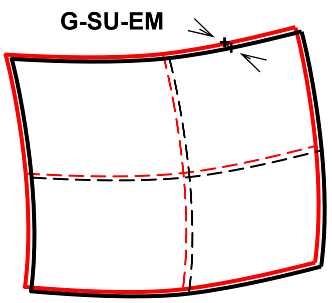
3.1.2.13 Embedded surfaces: G-SU-EM

Problem description: Set of surfaces where one completely overlaps the other(s). That is, one surface completely or partially includes the other. Set can include surfaces of any type.

Measurement: Whether there is a surface completely embedded within another surface within the designated accuracy.

Supporting information: Identical/Double elements unnecessarily increase the storage requirements and cancel out the consistency and validity of the original. They obstruct the handling of these models, e.g., the automatic creation of the topology. It is understood that elements that lie within one large one are also identical.

Recommendation: Delete double elements, thereby ensuring that “the required” element is retained.



Example: Embedded surfaces

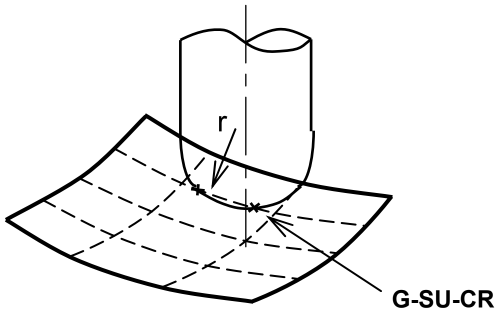
3.1.2.14 Surface with a small radius of curvature : G-SU-CR

Problem description: Surface has a small radius of curvature.

Measurement: Minimum radius of curvature, in any direction, on surface.

Supporting information: To guarantee the ability to modify a surface, create an offset surface and use the surface in downstream applications, the curvature radius of a surface must not fall short of a given minimum at any position or direction. The minimum acceptable curvature depends on the intended use of surface. If it will define an offset surface, for example, it must be large enough to prevent self-intersection of the offset surface. If the surface will eventually define a machined surface, then the minimum acceptable curvature must be large enough to prevent machining errors.

Recommendation: Surfaces that violate the given minimum curvature radius must be recreated, e.g., through approximation or smoothing.



Example: Surface with a small radius of curvature

3.1.2.15 Unused patches: G-SU-UN

Problem description: Surface has patches that are not used in part or in whole by any faces.

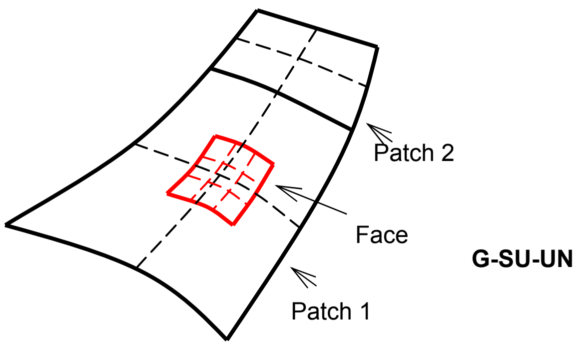
Measurement: Count of unused patches.

Supporting information: The area of a surface occupied by a bounded face can be so small that whole rows of the underlying surface patches are unoccupied. These unoccupied patch rows use up valuable storage space and generally can be erased without any problem.

Using this criterion, the surfaces that do not serve the purpose of defining bounded surfaces and therefore are most likely to be superfluous will also be found.

Sometimes the unoccupied face domains are still required in subsequent process steps. Their reconstruction is then time-consuming and only approximately possible.

Recommendation: If required, divide the surface along an appropriate patch border and delete the now-unused surfaces.



Example: Unused patches

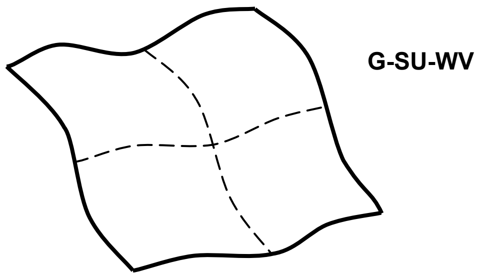
3.1.2.16 Wavy surface: G-SU-WV

Problem description: Surface has too many curvature sign changes.

Measurement: Count of curvature sign changes along any iso-parametric curve on the surface or patch.

Supporting information: An unintentional curvature within a surface is possibly critical for the styling, offset surfaces, NC processing, or other applications. There can also be problems internal to a CAD system.

Recommendation: Correct surface or regenerate with suitable fundamental conditions (degree, edge curves, or definition points).



Example: Wavy surface

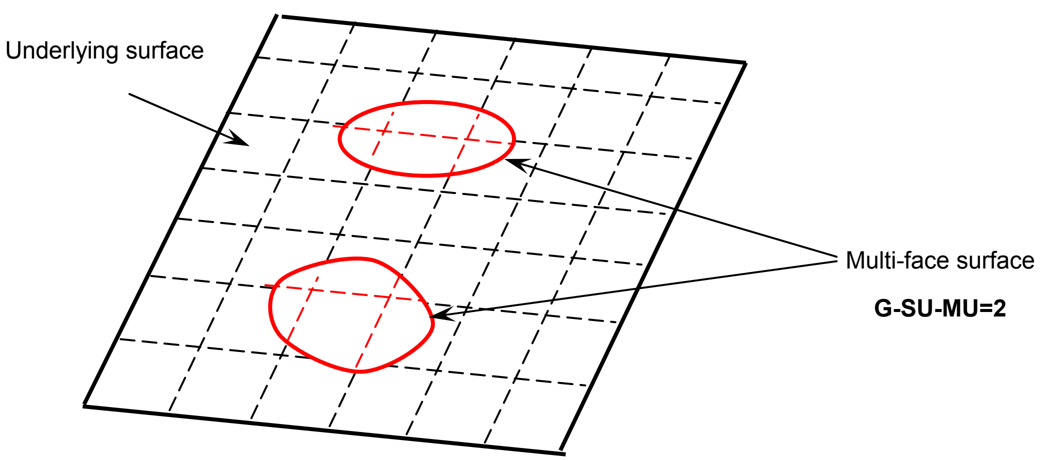
3.1.2.17 Multi-face surface: G-SU-MU

Problem description: A surface that is used (or referenced) by more than one face.

Measurement: Count of the number of faces that use this surface.

Supporting information: Several CAD systems and translators require a one-to-one relationship between surfaces and faces.

Recommendation: Create an independent surface for each face, preferably by splitting the existing surface.



Example: Multi-face surface

3.1.2.18 Folded surface: G-SU-FO

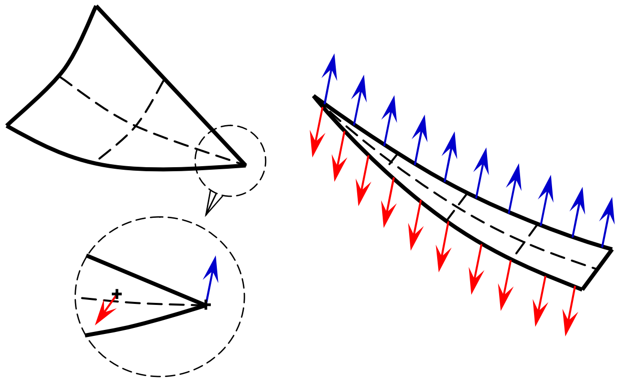
Problem description: Surface is folded in one or both parametric directions.

Measurement: Maximum angle between pairs of normal vectors in either parametric direction in a patch.

Supporting information: Generally, all normal vectors of a surface are shown uniformly facing the same direction, either into the component or out of it. Occasionally, deviations from this behaviour can occur at the edge of surfaces. This may cause problems in surface offset, in tooling path creation, or in other applications. For example, damage to the work piece can occur since the tool can cut into the surface, or rapid prototyping can result in incorrect objects.

A special case of a twisted surface close to its edge may be found at the tip of a “quasi” triangular patch. This is the case when two boundary curves, which are diverging upon a point, slightly project beyond the point of intersection.

Recommendation: Surfaces, where the vectors for normals have been turned around, should be newly created (under special consideration of the tangential conditions at the periphery). In the case where a vector at the tip of a triangular patch is flipped or turned around, the tip (within the bounds of admissible gaps and tiny elements) can be "cut off" so that the new, fourth edge of the patch receives an admissible length. Alternatively, a three-sided face with correct normals can be generated on the surface.



Example: Folded surface

3.1.2.19 Inappropriate degree planar surface: G-SU-ID

Problem description: A planar or nearly planar surface is defined with too high degree.

Measurement: Two measurements are necessary:

1. Whether the surface is planar or nearly planar: a surface is planar if, for a plane defined from a least-squares-fit of the surface, no point on the surface is further than the given tolerance from this plane. The given tolerance is a defined model value.
2. Whether the surface degree is too high: the highest polynomial degree of the u and v direction must be considered.

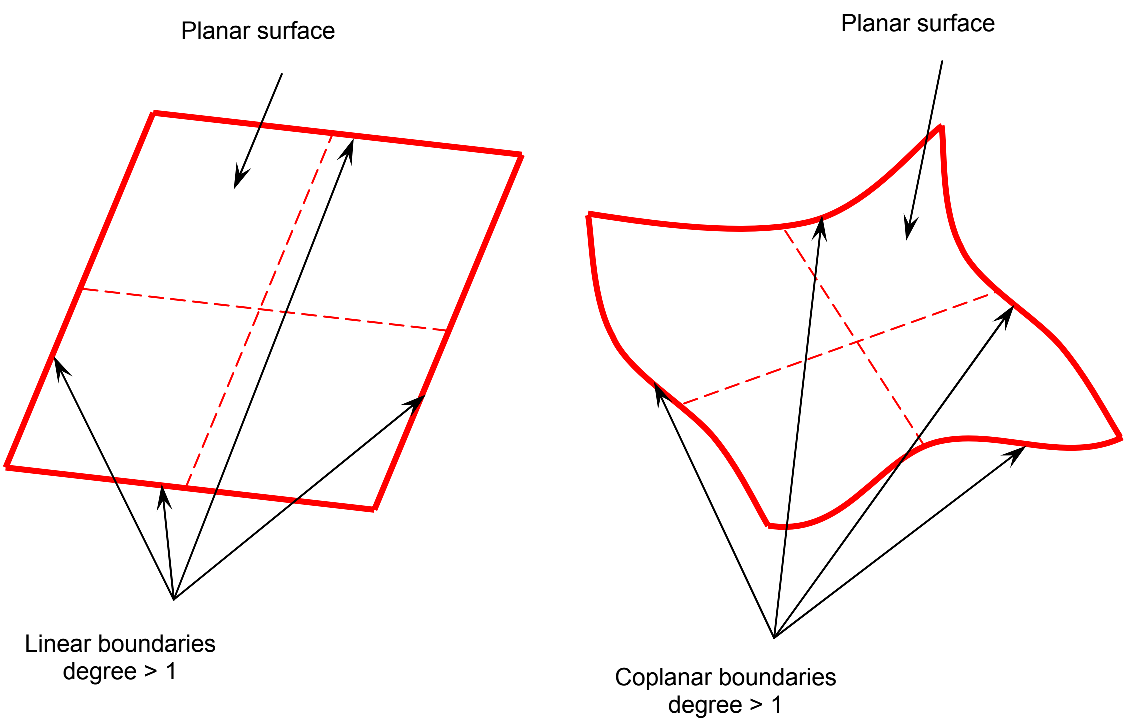
Supporting information: Over-specified polynomial surfaces can cause multiple errors when used as basic elements to build the digitalisation or when translated to other systems. When translated, the result might no longer be a planar surface and potentially an entity full of severe errors.

Recommendation: Several cases are possible:

- If the degree reduction to 1x1 doesn't modify the topology more than a given tolerance, then replace the polynomial plane with a plane or reduce the degree of the surface down to 1x1.
- If the degree reduction to 1x1 modifies the topology more than a given tolerance, then depending on the user willing:

  » Either rebuild as a planar face using a planar surface on the least square plane limited by the surface boundaries, or

  » Keep the surface as it is.



Example: Inappropriate degree planar surface

3.1.3 Edge

An edge is the boundary curve in the (*u*,*v*) parameter domain of a surface. Also an edge may refer to a topological element, which corresponds to a connection between one or two vertices.

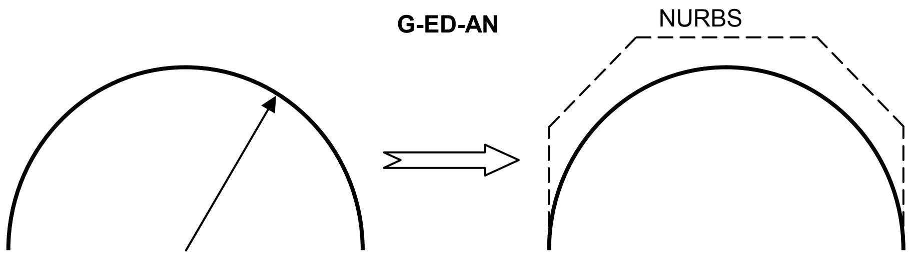
3.1.3.1 Analytical edge: G-ED- AN

Problem description: Some analytical edges cannot be translated into NURBS and, therefore, cannot be used by NURBS-based target systems.

Measurement: Whether the edge is analytical or not.

Supporting information: When an analytical edge is translated into NURBS, both the starting point and the ending point are defined, but these points will probably not meet the edge's end point. The system may cause errors when calculating a point on the edge.

Recommendation: All edges should be able to be translated to NURBS.



Example: Analytical edge

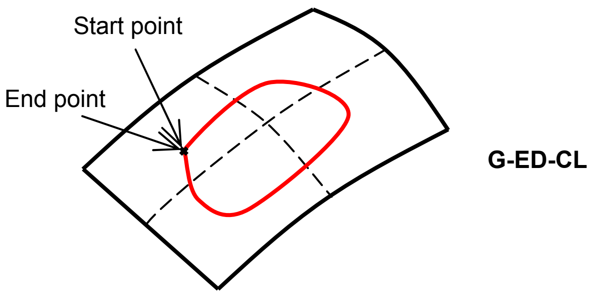
3.1.3.2 Closed edge: G-ED-CL

Problem description: Endpoints of edge are coincident.

Measurement: Whether edge endpoints are coincident.

Supporting information: Many current CAD systems build a circular or elliptic edge by splitting it into two half-edges to avoid the problems of coincident edge end points. Although there are some CAD systems that do not use such an approach internally, there is the risk of unwanted behaviour (in search algorithms, inconsistencies) in many other CAD systems after model exchange via interface programs.

Recommendation: Divide edge into parts, if necessary.



Example: Closed edge

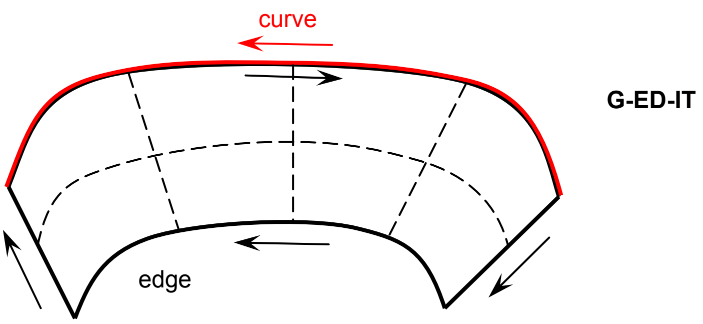
3.1.3.3 Inconsistent edge on curve: G-ED-IT

Problem description: Parametric direction of edge is inconsistent with its curve.

Measurement: Whether edge direction of the edge in parameter space is consistent with curve direction of corresponding 3D curve.

Supporting information: It may happen that both directions are opposite, depending on the basic CAD system considered. In such a case, the face edge curve definition is not fully correct, and problems may arise in data exchange. Note that this problem is not related to the problem G-LO-IT (3.1.4.2 Inconsistent edge in loop: G-LO-IT), where the direction of an edge is inconsistent to the direction of the loop.

Recommendation: Correct the direction of 3D curve, if necessary.



Example: Inconsistent edge on curve

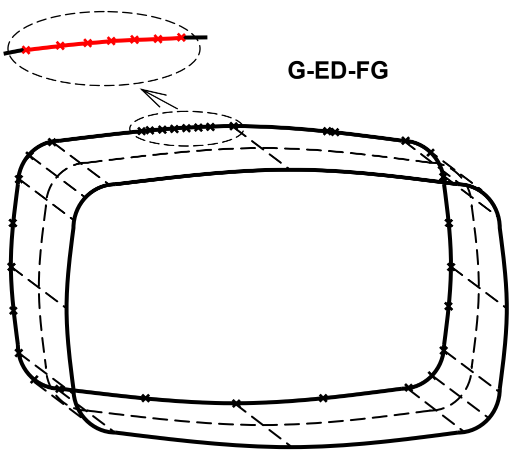
3.1.3.4 Fragmented edge: G-ED-FG

Problem description: Portion of underlying curve used by edge is defined by too many segments. This criterion does not apply when an edge is purely a topological entity.

Measurement: Count of partial or complete curve segments within trimmed portion of edge.

Supporting information: A disproportionately large number of segments within a boundary curve raises the risk of tiny elements, as well as discontinuity, and impedes making changes or creating new geometric elements like profile surfaces composed of such fragmented edge curves.

Recommendation: Correct boundary curves by approximation within the given accuracy and recreate the bounded surface with the improved edge.



Example: Fragmented edge

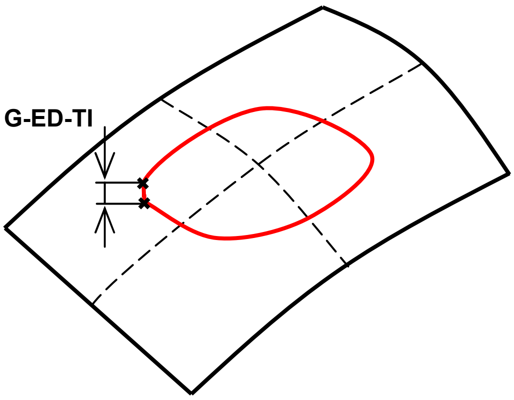
3.1.3.5 Tiny edge: G-ED-TI

Problem description: Overall extent of edge is below a given value.

Measurement: Length of edge or curve arc.

Supporting information: Edge curves that fall short of a particular length may lead to invalid elements, especially during the exchange of data (in a system with reduced precision). Because of this, the definition of bounded surfaces, as well as the correct topology information, may get lost so that only the untrimmed surface will be transferred.

Recommendation: Consolidate the edge curves with bordering edge curves for a new bounded surface, or delete/enlarge the tiny edge curves and rectify the connecting elements appropriately.



Example: Tiny edge

3.1.4 Edge Loop

An edge loop (in some CAD systems called: *domain*) is the closed set of consecutive edge curves that limit a bounded surface (face), projected onto the underlying surface. An edge loop must comply with several quality criteria that are closely related to the criteria for edge curves.

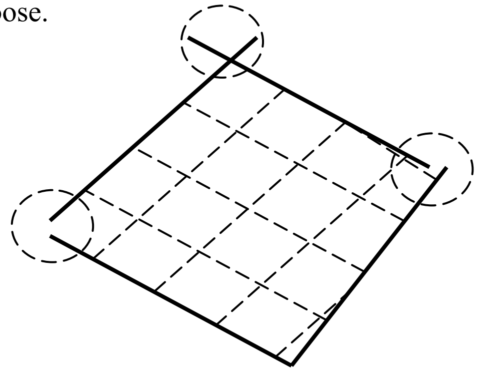
3.1.4.1 Large edge gap (G_0 discontinuity): G-LO-LG

Problem description: A large distance between or overlapping of endpoints of adjacent edges—a G_0 discontinuity.

Measurement: Distance between edge endpoints at common vertex.

Supporting information: In the case of a discontinuity of boundary curves, gaps and overlapping of their segments lead to difficulties during the definition of the bounded surface that may result in a loss of face definition during data exchange and a transfer of the untrimmed surface to the other system.

Recommendation: Redefine the ends of boundary curves to each other within the tolerances for identical points. Thereby, the adaptation of the curve end is preferential to inclusion of tiny segments with the risk of unwanted noose.



G-LO-LG

Example: Large edge gap (G_0 discontinuity)

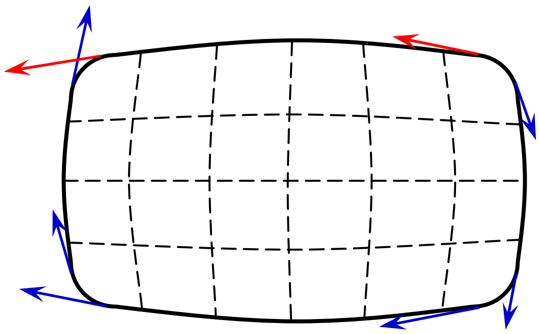
3.1.4.2 Inconsistent edge in loop: G-LO-IT

Problem description: Parametric direction of edge is inconsistent with its direction in loop.

Measurement: Whether directions of edge and loop are consistent.

Supporting information: Boundary curves that are not flowing in the loop direction may lead to unwanted self-penetration and face degeneration after data translation to some other systems.

Recommendation: If necessary, reverse the unwanted edge direction and recreate the bounded surface.



G-LO-IT

Example: Inconsistent edge in loop

3.1.4.3 Self-intersecting loop: G-LO-IS

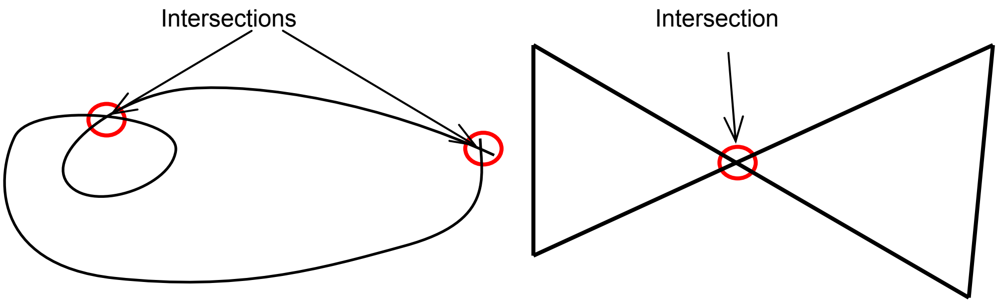
Problem description: Loop intersects itself at a location other than endpoints.

Measurement: Whether one or more intersection points exist in the loop within the designated accuracy.

Supporting information: The correct face definition is violated if the edges of the loop show a self-intersection.

Recommendation: See self-intersecting curve recommendation (3.1.1.6 Self-intersecting curve : G-CU-IS).

G-LO-IS



Example: Self-intersecting loop

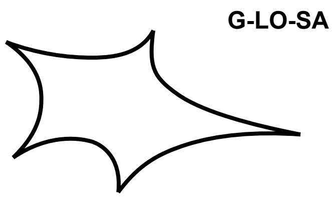
3.1.4.4 Sharp edge angle: G-LO-SA

Problem description: Angle below a given value between adjacent edges within a loop.

Measurement: Angle between edge tangent vectors at common vertex.

Supporting information: A sharp edge angle should not be interpreted as an unwanted self-approximation within a loop.

Recommendation: No edge changes are possible if the design intent requires the sharp angle. This may, however, be an intended design feature to avoid a sharp boundary angle in a surface corner, which may be more dangerous for the correct face definition.

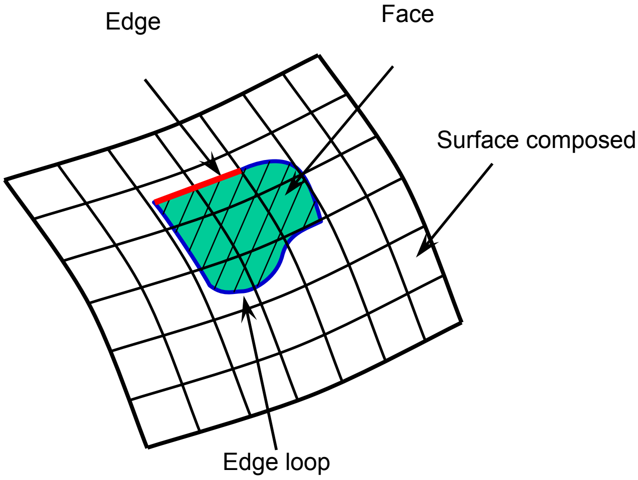


Example: Sharp edge angle

3.1.5 Face

Surfaces are used to define faces through bounding contours (a bounded surface) generally described as mathematical boundary curves on the surface. A “Bounded surface“ or simply “Face“ defines the geometrical and topological surface element of a surface structure (shell). It may consist of the underlying surface that forms the mathematical basis, together with the boundary curves (loop) that are projected upon it and, if necessary, with edges that limit the features like holes, indentations/ recesses. It is understood that the outer boundary curve (loop) of a face is a closed G_0 continuous curve.

The association between the underlying surface and the face makes clear that a great deal of quality criteria are principally applicable for both and are not repeated here (polynomial degree, curvature, internal continuity, tiny elements, and identical elements). Additionally, some further criteria are applicable for the relationship between edge curves and the bounded surface.



Example: Surface with face

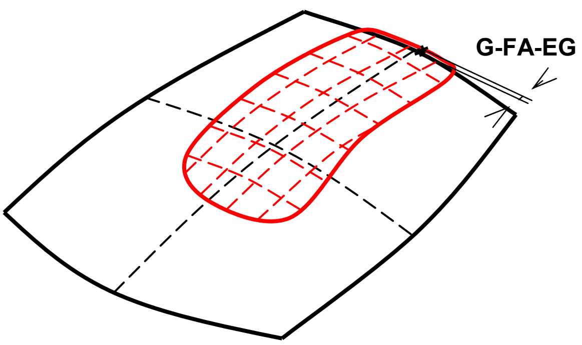
3.1.5.1 Large edge face gap: G-FA-EG

Problem description: Distance between an edge and the surface that it trims is above the given accuracy.

Measurement: Maximum distance between each point on edge and corresponding location on the surface.

Supporting information: Edges defined too far from the surface (normal or laterally) prevent the correct definition of the face. They need to be projected onto the surface with greater precision.

Recommendation: Create curves that are always within the range of tolerances of identical elements, as sectional curves or projections or, where necessary, regenerate or re-project the curve.



Example: Large edge face gap

3.1.5.2 Large vertex gap: G-FA-VG

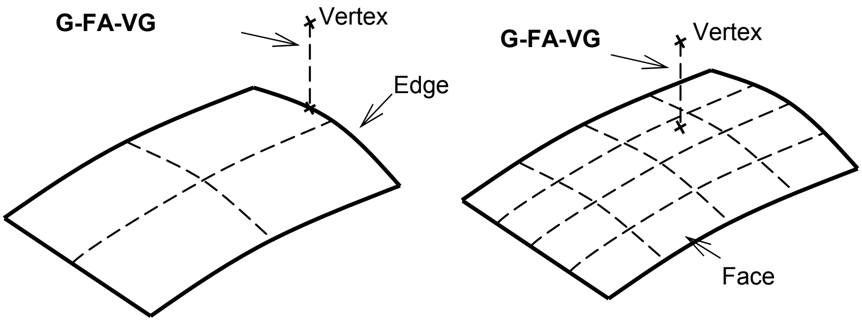
Problem description: Distance between a vertex and its corresponding edge or face that it trims is greater than a given value.

Measurement: Maximum distance between vertex point and its associated edge endpoint or face that it trims.

Supporting information: B-rep solids consist of the topological elements Vertex, Edge, and Face, which are assigned to the geometrical elements Point, Curve, and Bounded surface.

The point that corresponds to a Vertex must lie within a stipulated accuracy on the associated edge and bounded surface. If the distance between the point and the edge or face exceeds this value, then the solid is said to have a large vertex gap.

Recommendation: If possible, project the point onto the curve; otherwise, regenerate. If possible, project the point onto the face; otherwise, regenerate.



Example: Large vertex gap

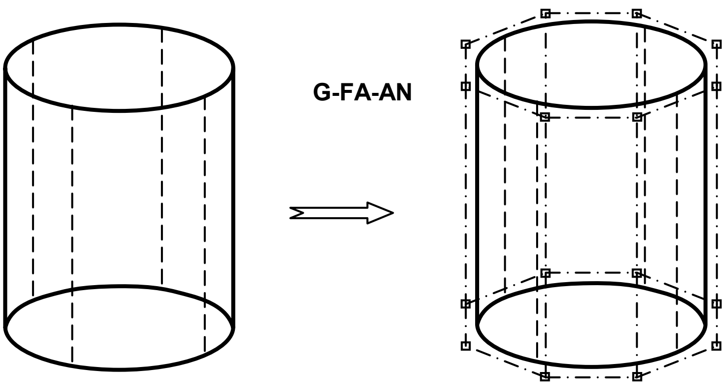
3.1.5.3 Analytical face: G-FA-AN

Problem description: Some analytical faces cannot be translated into NURBS and, therefore, cannot be used by NURBS-based target systems

Measurement: Whether the face is analytical or not.

Supporting information: When an analytical face is translated into NURBS, both the starting boundary and the ending boundary are defined, but these boundaries will probably not meet the face's edge. The system may cause errors when calculating a boundary of the face.

Recommendation: All faces should be able to be translated to NURBS.



Example: Analytical face

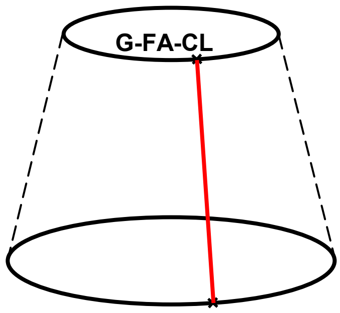
3.1.5.4 Closed face: G-FA-CL

Problem description: One or both pairs of opposite face boundaries are coincident.

Measurement: Whether a face is open or closed topologically and not geometrically.

Supporting information: This design may be standard case in some CAD systems (examples: cylinder, torus) but may cause problems in data exchange. There are other systems that avoid such a design by splitting into half elements.

Recommendation: Systematically split face into half elements.



Example: Closed face

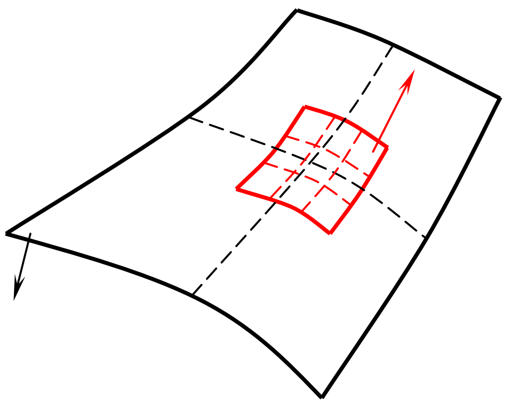
3.1.5.5 Inconsistent face on surface: G-FA-IT

Problem description: Direction of face normal is inconsistent with surface.

Measurement: Whether the direction of the normal vectors of the face and its underlying surface is consistent.

Supporting information: In some CAD systems a face has no own normal orientation of its own but has the same normal orientation as the underlying surface. Then the directions of face and surface are automatically consistent.

Recommendation: Rebuild the face in order to get a consistent normal direction.



Example: Inconsistent face on surface

3.1.5.6 Intersecting loops: G-FA-IS

Problem description: Pair of loops in the same face that intersect each other.

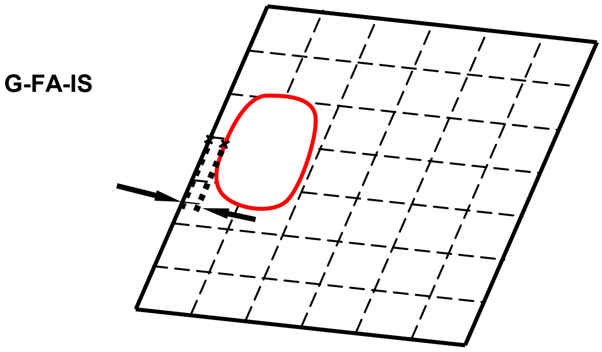
Measurement: Whether one or more intersection points exist between two loops in the same face within the designated accuracy.

Supporting information: Penetration or contact of edge loops caused by using values smaller than the minimum distance accuracy can lead to invalid faces (loss of face definition) and to loss of integrity of a topology.

This criterion covers the penetration or contact between an outer edge loop with an inner edge loop or between two inner edge loops.

Accuracy values should be similar to those for G-LO-IS (3.1.4.3 Self-intersecting loop: G-LO-IS).

Recommendation: Enlarge the space between edge loops, remove loops and, where necessary, partition faces or consolidate edge loops while maintaining design intent.



Example: Intersecting loops

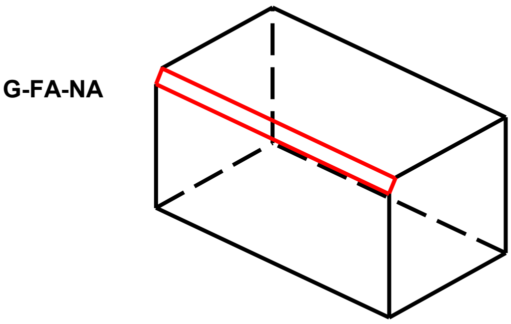
3.1.5.7 Narrow face: G-FA-NA

Problem description: Face is consistently too narrow in **one** direction.

Measurement: Maximum width of face in narrow direction.

Supporting information: Faces that fall short of a particular dimension can lead to invalid elements and thereby to gaps, especially with certain geometrical operations (e.g., scaling formation or offsets), during the exchange of data (in a system with reduced precision) or by subsequent processing (FEM or NC). Reworking these elements requires a considerable increase in effort. These elements may occur unintentionally through filleting.

Recommendation: Delete minimal bounded surface or enlarge and adapt the neighbouring elements accordingly.



Example: Narrow face

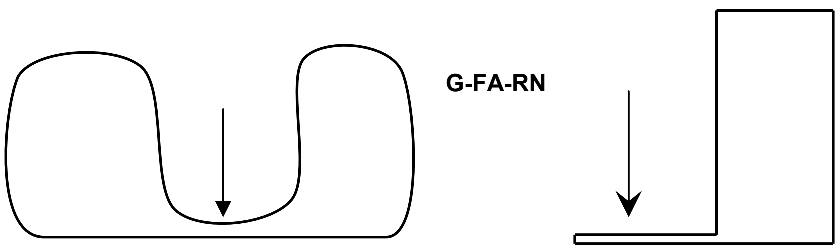
3.1.5.8 Narrow region: G-FA-RN

Problem description: Portion of a face that is too narrow compared to a given value.

Measurement: Width (proximity) between the two closest points in a loop or between two loops in the same face as well as the length of the narrow region.

Supporting information: The correct face definition is violated if there is close approximation between two or more loops.

Recommendation: Split the face in order to keep the desired part and delete the narrow region. Ensure continuity within the redesigned geometry.



Example: Narrow region

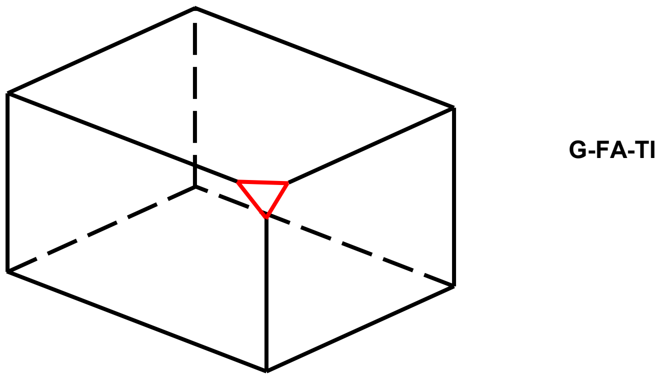
3.1.5.9 Tiny face: G-FA-TI

Problem description: Overall extent of face is too small.

Measurement: Surface area of tiny face, compared to the given accuracy.

Supporting information: Faces that fall short of a particular dimension can lead to invalid elements and thereby to gaps, especially with certain geometrical operations (e.g., scaling formation of offsets), during the exchange of data (in a system with reduced precision) or by subsequent processing (FEM, STL, visualisation, or NC). Reworking these elements requires a considerable increase in effort. These elements may occur unintentionally through filleting.

Recommendation: Delete minimal bounded surface, or enlarge and adapt the neighbouring elements accordingly.



Example: Tiny face

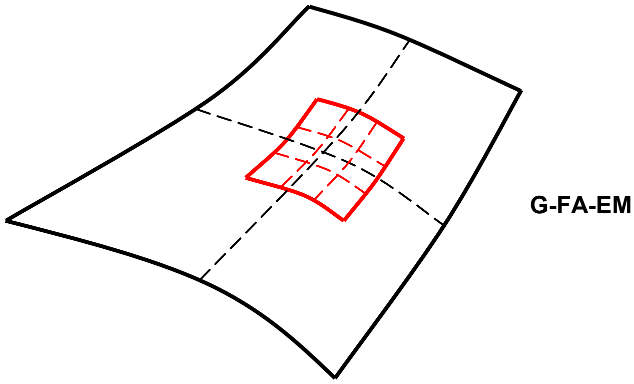
3.1.5.10 Embedded faces: G-FA-EM

Problem description: Set of faces where one completely overlaps the other(s). Set can include faces of any type.

Measurement: Whether there is a face completely embedded within another face within the designated accuracy.

Supporting information: See G-SU-EM (3.1.2.13 Embedded surfaces: G-SU-EM).

Recommendation: Delete the appropriate face.



Example: Embedded faces

3.1.6 Shell

A shell is a set of sewn faces. Neighbouring bounded surfaces, which together form a particular part or complete surface of an object, are called composite surfaces/surface groups or topology. Within a topology, special requirements apply regarding the quality of faces.

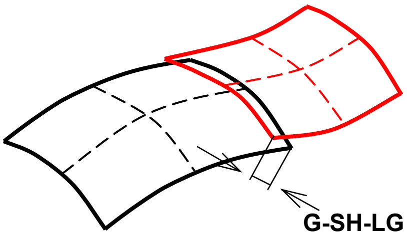
3.1.6.1 Large face gap (G_0 discontinuity): G-SH-LG

Problem description: Large distance between or overlapping of adjacent faces—a G_0 discontinuity.

Measurement: Maximum distance between pairs of nearest points on each face along common edge.

Supporting information: Position continuity, i.e., coincident position within the given accuracy of bounded surfaces within a topology is the most important quality characteristic within every surface group. A permissible discontinuity that is within the bounds of the accuracy can lead to a loss of the topology in the case of a change in the system or in the range of tolerances or can cause some systems to perform an automatic correction (Healing). Because of this, unintentional changes or new (tiny) elements can occur.

Recommendation: In the case of gaps by face transitions, regenerate the affected faces with common boundary curves. **Note:** Tangential and curvature continuities must be maintained as they existed before the correction.



Example: Large face gap
(G_0 discontinuity)

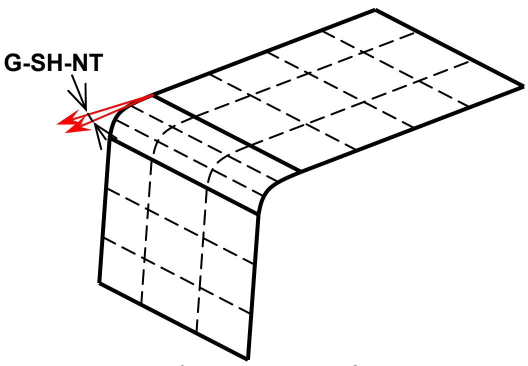
3.1.6.2 Non-tangent faces (G_1 discontinuity): G-SH-NT

Problem description: Non-tangent angle between adjacent faces—a G_1 discontinuity.

Measurement: Maximum angle between surface normals evaluated at pairs of nearest points on each face along common edge (G_0 continuous).

Supporting information: Tangent continuity may be needed for styling, CNC milling, or casting.

Recommendation: Regenerate the affected faces with appropriate boundary conditions.



Example: Non-tangent faces
(G_1 discontinuity)

3.1.6.3 Non-smooth faces (G_2 discontinuity): G-SH-NS

Problem description: Large curvature change between adjacent faces—a G_2 discontinuity.

Measurement: Curvature continuity at the contact point of two faces (by a given position/tangential continuity) means:

a) Check Curvature continuity in consecutive normal section planes.

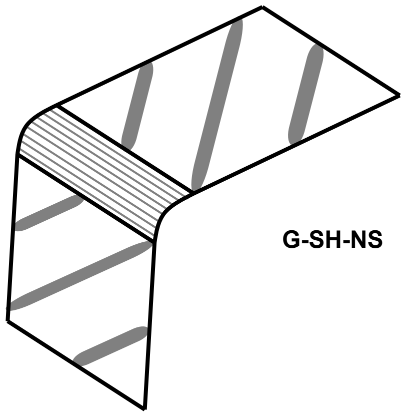
b) Central points of curvature radii lie on same side of the faces.

c) Difference of absolute values of radii, divided by mean value of radii, is below the given accuracy,,that is:

$$G - SH - NS = \frac{2|r_1 - r_2|}{|r_1| + |r_2|}$$ (note: G-SH-NS is always positive)

Supporting information: Curvature continuity may be needed for high speed milling or design surfaces.

Recommendation: Regenerate the affected faces with appropriate boundary conditions.



Example: Non-smooth faces (G_2 discontinuity)

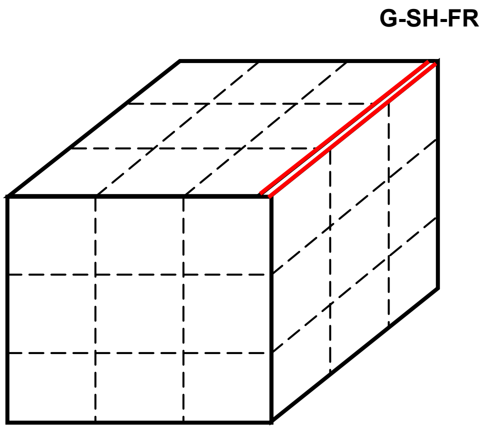
3.1.6.4 Free edge: G-SH-FR

Problem description: A free edge is used by only one face within a shell.

Measurement: Whether edge is used by two faces.

Supporting information: Shell has free edges that are not sewn together. This is not usable, for example, for trimming operations. Free edges may be intentional and require user interpretation. Examples of intentional free edges are outer boundaries, or holes within an open shell.

Recommendation: Recreate shell to eliminate undesirable free edges.



Example: Free edge

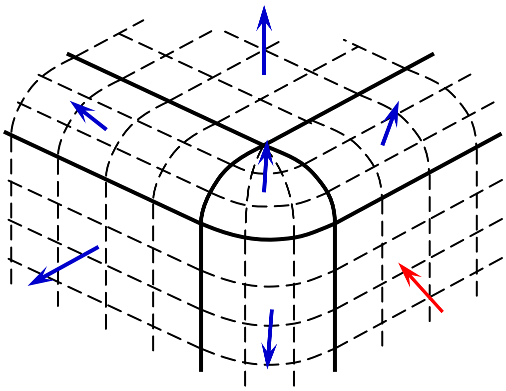
3.1.6.5 Inconsistent face in shell: G-SH-IT

Problem description: Adjacent faces with opposite normals along their common boundary. It may, however, happen that the shell normal orientation is inconsistent with the face normal orientation. (This may cause problems in data exchange to Virtual Reality (VR) systems when the shell element is not transferred. It may leave only the single face elements with their possibly inconsistent orientation.)

Measurement: Whether normals are identical along common boundary.

Supporting information: Uniform orientation of the face normals within a topology is necessary. (Examples of data uses where orientation of normals can lead to problems are the determination of machining direction for milling and shaded visualisations.)

Recommendation: Where necessary, invert individual face normals so that all face normals are topologically uniformly oriented, i.e., “away from the material.”



G-SH-IT

Example: Inconsistent face in shell

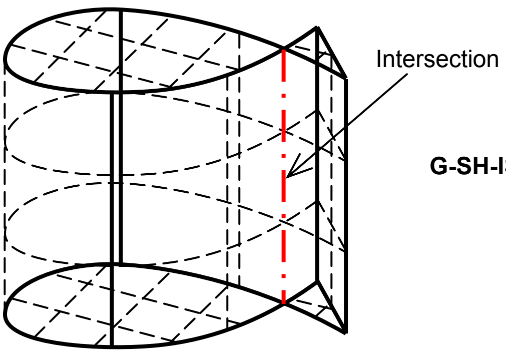
3.1.6.6 Self-intersecting shell: G-SH-IS

Problem description: Shell intersects itself.

Measurement: Whether any faces of the shell intersect at locations other than the edges within the given measurement accuracy.

Supporting information: Self-intersecting shells must never occur in a model because they are impossible to manufacture. Self-intersections of shells can result, for example, if a curve is extruded along a tight corner. (See 3.1.2.8 Self-intersecting surface: G-SU-IS)

Recommendation: Check design intent to eliminate self intersection.



G-SH-IS

Example: Self-intersecting shell

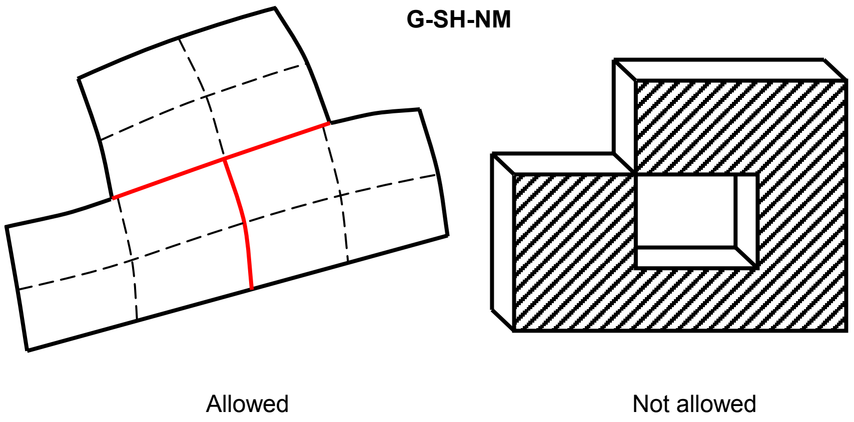
3.1.6.7 Over-used edge: G-SH-NM

Problem description: Edge is used by more than two faces. For solids, this is also known as "non-manifold solid brep."

Measurement: Whether edge is used by more than two faces.

Supporting information: For the topological explicitness of a surface, every inner face edge must have one explicit neighbouring face, i.e., may not have more than one neighbouring edge and therefore is free from bifurcation/junctions. It is, however, acceptable for a face edge to border on several neighbouring face edges, one after the other ("T-type butt joint").

Recommendation: Remove or redefine violating faces.



Example: Over-used edge

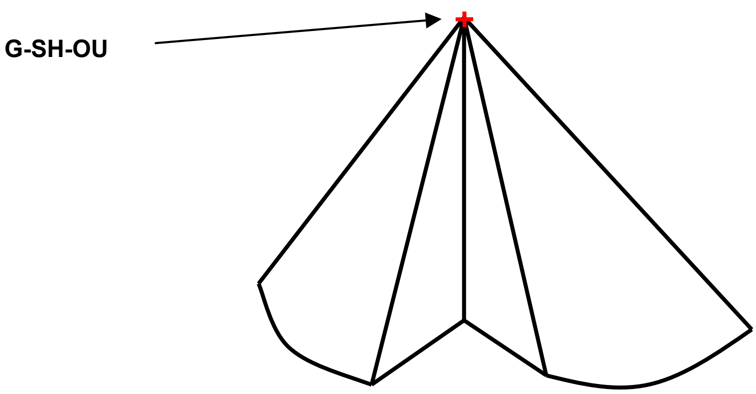
3.1.6.8 Over-used vertex: G-SH-OU

Problem description: Vertex is used by too many edges.

Measurement: Count of edges using the vertex.

Supporting information: This is a warning more than a specific requirement. While such a situation may be acceptable, too many edges at one vertex is often an indication that problems exist.

Recommendation: Check design intent.



Example: Over-used vertex

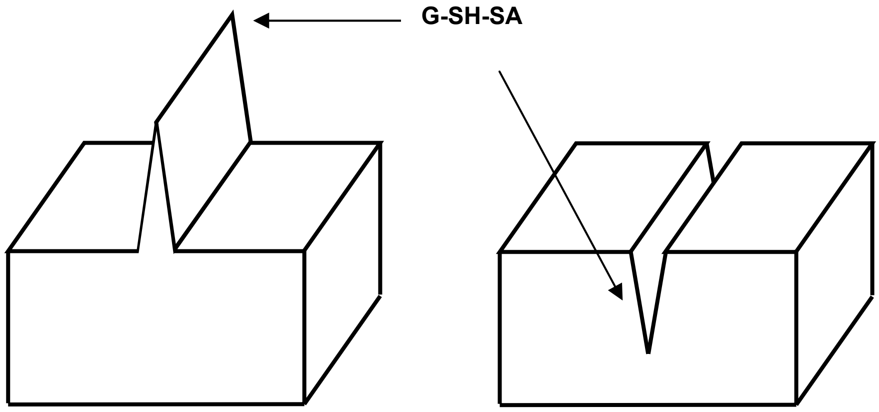
3.1.6.9 Sharp face angle: G-SH-SA

Problem description: Extreme angle between adjacent faces.

Measurement: Maximum angle between pairs of face normal vectors along common boundary.

Supporting information: Sharp face angles occur when the absolute value of the angle between the faces approaches 180 degrees. Such areas are not realistic and cannot be produced. They arise, for example, through subtraction of a cylinder from a cube.

Recommendation: Check the design and reconstruct as necessary.



Example: Sharp face angle

3.1.7 Solid

Solids consist of one or more closed shells that enclose a volume. In most CAD systems, solids are the preferred method of representation. Therefore, a solid model can be defined as a complete representation of a product shape, and points of its interior are all connected. Every point can be classified as inside the boundary of a solid, outside the boundaryh, or on the boundary.

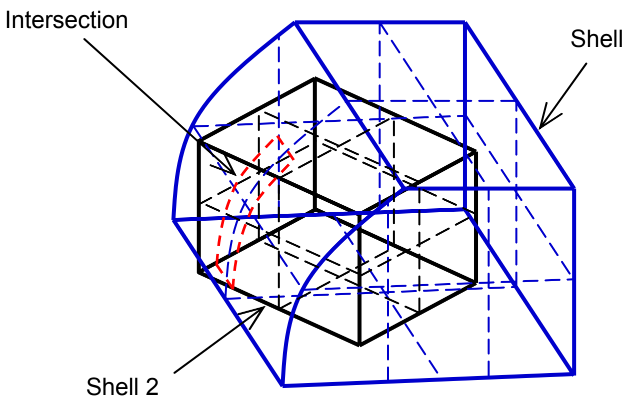
3.1.7.1 Intersecting shells: G-SO-IS

Problem description: Pair of shells in a solid that intersect each other.

Measurement: Whether any faces of different shells intersect at locations other than the edges.

Supporting information: Intersecting shells occur if, for example, a blend is applied to the outside of a thin-walled solid.

Recommendation: Check design intent to eliminate intersection.



G-SO-IS

Example: Intersecting shells

3.1.7.2 Multi-volume solid: G-SO-MU

Problem description: Solid has more than one distinct volume.

Measurement: Whether solid has only one distinct volume.

Supporting information: In the case of a number of CAD systems, solids can consist of several bodies; i.e., a solid consists of a collection of at least two disjunctive bodies (not touching each other). These so-called multi-body solids cannot be handled by all CAD systems and are therefore to be avoided.

Recommendation: The individual bodies should, in each case, be converted into an individual solid, e.g., in that one cancels/undoes the unification operation. Afterwards, one solid will exist per body. This will occur automatically during a transfer via STEP.

3.1.7.3 Embedded solids: G-SO-EM

Problem description: Set of solids where one completely contains the other(s).

Measurement: Whether one solid completely contains the other.

Supporting information: Redundant solids make the model unnecessarily complex and might lead to wrong interpretations.

Recommendation: Remove unused solid.

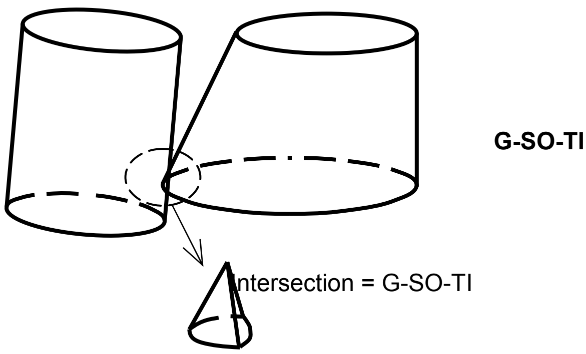
3.1.7.4 Tiny solid: G-SO-TI

Problem description: Overall extent of solid is too small.

Measurement: Volume of the solid.

Supporting information: Solids that fall short of a particular dimension in two directions in space should be avoided. Depending on the interface and the system internal parameter for the degree of accuracy, these elements can cause problems or be lost during the exchange of data. Often these elements may also occur unintentionally during the modelling process (i.e., intersection of two solids that only slightly penetrate each other) and cannot be produced.

Recommendation: This error source can be eliminated through displacing or enlarging the affected elements. If appropriate, remove tiny elements entirely, or eliminate them by enlarging the neighbouring elements and then deleting the tiny elements.



Example: Tiny solid

3.1.7.5 Solid void: G-SO-VO

Problem description: Unintentional internal cavity within a solid defined by an interior shell.

Measurement: Whether solid has only one exterior shell.

Supporting information: A solid should not include any unwanted cavities. Cavities often occur unintentionally during modelling. Unintentional voids can make the solid unnecessarily complex and increase the amount of data. Also, intentional cavities could possibly be irrelevant for the recipient of the data (for model section analysis, for example).

Recommendation: Critically check solids to determine whether any unwanted cavities are present and delete where appropriate.

3.2 Non-Geometric Quality Criteria Descriptions

This chapter includes quality criteria related to the non-geometric aspects of a CAD model, also known and discussed as “organisational criteria,” which include the model structure criteria.
The model structure is an essential prerequisite for the clarity and usability of a CAD 3D data model. It also allows the safe and speedy reduction of the model contents to a useful exchange scope.
A model structure respects the following characteristics:

- It must be recognisable, comprehensible, and firmly allocated to the CAD data model.
- It should be able to differentiate between auxiliary geometry and essential product geometry (i.e., wire, face, and solid geometry).
- It should be able to differentiate between right/left-handed and non-handed parts.
- It should be able to reproduce logical relationships such as functions, assemblies, or similar.
- It should be able to differentiate between changeable and non-changeable contents.
- It must be created and used in accordance to the rules concerning data quality documentation during the exchange of said data.

3.2.1 CAD model

3.2.1.1 Non-standard CAD version: O-CM-CV

Problem description: The version of the CAD system used to store the CAD part is not compatible (too old or new) with the current CAD system version. Errors might occur such as inability to open, modif, or convert parts.

Measurement: Check the version of the CAD system with which the part was last saved regarding the company standard.

Supporting information: When a part that has been saved on an incompatible (too old or new) version of a CAD system into the current version, data loss or invalid recovery could result. “Version” is a general term that is used to identify an updated or modified level of software, which includes a version (Ver), release (Rel), revision (Rev), and a service pack (SP).

The “standard CAD version.” as well as the dates to change this, should be agreed on between related companies and announced as early as possible.

Recommendation: Store parts compliant to the company standards.

3.2.1.2 Wrong CAD startup environment: O-CM-SE

Problem description: In most CADsystems there are some global parameters, e.g., pattern settings, defining the properties for all parts generated with this environment. If a part has been created in a wrong environment that does not belong to the company standard, this may lead to loss of information in the receiving system (e.g.. different pattern).

Measurement: Check whether the environment settings are according to the company standard.

Supporting information:

Recommendation: Use environment settings according to the company standard.

3.2.1.3 Non-standard accuracy parameter: O-CM-AP

Problem description: The accuracy parameter is directly linked with the mathematical accuracy and representation of the part, e.g., tolerance for identical curves, intersection projection, and so on. Surfaces as well as topologies can become inconsistent for further process of the part, if the original value of the accuracy parameter does not comply to the company standard.

Measurement: Check whether the accuracy parameter is according to the company standard.

Supporting information: In most CADsystems, the accuracy parameters have an impact on the geometrical tolerances of the elements. The most common accuracy parameters are as follows:

- Large gap accuracy parameter: The maximum tolerable distance that is defined as “neighbouring” if there is a gap in precise terms
- Non-tangent accuracy parameter: The maximum tolerable angle at which a neighbouring curve, edge, surface, or face is defined to be smoothly connected even if they are broken in precise terms.
- Tiny accuracy parameter: The minimum tolerable length that is not defined as a tiny element.

Recommendation: Use the accuracy parameters according to the company standard.

3.2.1.4 Hybrid model: O-CM-HY

Problem description: Model contains a mixture of geometric entity types and representations: solids, open shells, faces, edge loops, edges, surfaces, curve, or points.

Measurement: Whether all entities of a lower order are derivatives of the higher order geometry type (e.g., curves are, in fact, the edge loop of a boundary representation).

Supporting information: See Section 1.4 for information on the relative importance of the different kinds of data that may appear in a hybrid model.

Recommendation: Where preferred business practices are contrary to these criteria, it is important to convey specific directions as part of the product data (i.e., drawing dimensions supersede 3D geometry.)

3.2.1.5 Multi-solid model: O-CM-MU

Problem description: Model contains more than one solid.

Measurement: Count of solids in model.

Supporting information: A part is defined here as a CAD file at the operating system level. A number of CAD/CAM systems cannot handle several solids in one part but expect in each case only one solid per part. This can lead, for example, to problems during the exchange of data should one want to transfer complete assemblies.

Recommendation: In each case, store the individual solids in a separate part.

3.2.1.6 Special character used in CAD model name: O-CM-SC

Problem description: Special characters can be used in the name of parts. The use of special characters might cause problems during the exchange of data between CAD systems.

Measurement: Search for special characters in the CAD model name.

Supporting information:

Recommendation: Use only characters:A-Z, 0-9, and _. Do not use national characters or dieresis and special characters like $, %, &, ##…

3.2.1.7 Non-standard item name: O-CM-IN

Problem description: Item is the general term for an object or group of objects in the CAD area. Items might be parts, assemblies, drawings, etc.

In the area of PDQ, the most often used item name is a CAD model name. Most companies, especially OEMs, have fixed specific naming conventions in CAD guidelines to ensure information quality and (automated) usability in receivers' applications and processes. Not meeting these conventions might lead to a break in automated processes.

Measurement: Whether an item name complies with company standards, as well as the naming rules set forth by the CAD specifications

Supporting information: Compliance with the naming rules set forth by the CAD specifications is prerequisite to data flow. In addition, the item name must further meet company standards agreed upon with customers as appropriate.

For example, check if an item name is composed of lowercase alphanumeric characters, "_," and "-," does not exceed 63 bytes in length, and further conforms to relevant naming rules such as a "product part number (3 bytes) + part name (8 bytes) + clerk ID (8 bytes), and control number (6 bytes)" format requirement.

Recommendation: Replace the item name by a name consistent with the company standard.

3.2.1.8 Non-standard physical file name: O-CM-PN

Problem description: If the physical file name is not correct, the inability to load data results in an error.

Measurement: Whether an item name complies with company standards agreed upon with customers as appropriate, as well as the naming rules set forth by the CAD specifications

Supporting information: Compliance with the naming rules set forth by the CAD specifications is prerequisite to data flow. In addition, the item name must further meet company standards agreed upon with customers as appropriate. It is necessary, for example, to check to see if an item name is composed of lowercase alphanumeric characters, "_," and "-," does not exceed 63 bytes in length, and further conforms to relevant naming rules, such as a "product model number (3 bytes) + part name (8 bytes) + clerk ID (8 bytes), and control number (6 bytes)" format requirement.

Recommendation: Replace the physical file name consistent with the company standard.

3.2.1.9 Too large physical file size: O-CM-FS

Problem description: The size of a part file (e.g., in Kilobytes) extends a given limit. Such large files might cause problems in transmission and handling in the receiving system (disc space problems) or opening it (memory overflow).

Measurement: Size value(s) (e.g., in Kilobytes) of a physical file.

Supporting information: The part size has a direct influence on performance.

Some CAD systems might have more than one parameter to control the file size.

Recommendation: Delete unnecessary entities, replace (approximate) space consuming entities, or split part if applicable.

3.2.1.10 Non-standard item property: O-CM-IP

Problem description: Some systems use attributes or parameters on part level. They typically contain organisational information and are used for automation in PDM and CAE environments. Missing part attributes or wrong values can disturb automated processes.

Measurement: Check if all required item properties are present and have an admissible value.

Supporting information: Item-specific information rarely has a direct impact on parting work but may contain essential parameters for running applications such as those for automating design and PDM entry sequences, and application execution. Hence, formulating rules and complying with them are important. Some CAD systems do not provide functions to manage those properties and a company might decide to use some kind of text to declare property information instead.

Recommendation: Ensure that the item property and its value are consistent with the company standards.

3.2.1.11 Item data consistency incorrect: O-CM-IC

Problem description: While creating a part, CAD systems might create internal inconsistencies. Those inconsistencies can create problems among the whole lifecycle of the part (modifications, data exchange, upgrade).

Measurement: Whether the check made by the part consistency verification feature specific to the CAD system has been verified.

Supporting information: Most CAD systems have an internal function that enables the detection of critical file inconsistencies.

Recommendation: Be sure to save parts only if they have been verified valid.

3.2.1.12 Non-standard reference set: O-CM-RS

Problem description: A reference set is used for structuring the elements within a part. Using non-standard reference set may lead to design, analysis, or machining errors.

Measurement: Check whether the current reference set is compliant with the company specific reference set.

Supporting information: A reference set is part of a part and can be referenced by an assembly.

A reference set is used to hide auxiliary graphics in an assembly or reference specific representations (such as facets). A standardised reference set facilitates assembly creation. The loss of a reference set leads to confusion, requiring repeated machining.

A reference set that contains a simplified part can save the time spent loading large assemblies.

Recommendation: Use a reference set as defined by company standards.

3.2.1.13 Encapsulated entities used: O-CM-EE

Problem description: Encapsulated entities may not be correctly reproduced during data conversion.

Measurement: Check whether encapsulated entities exist in the part.

Supporting information: Encapsulated entities can be used to define a shape or standard elements (screws, bearings, electrical connectors,…) for multi-instantiation purposes.

Recommendation: Do not use encapsulated entities or replace them by the original element according to the company standard.

3.2.1.14 Unused encapsulated entities present: O-CM-UP

Problem description: An unused encapsulated entity is defined but not referenced in the CAD part. It has no addedvalue.

Measurement: Check whether unused encapsulated entities exist in the part.

Supporting information:

Recommendation: Delete unused encapsulated entities.

3.2.1.15 Identical encapsulated entity: O-CM-IE

Problem description: A CAD system might allow managing encapsulated entities with names relative to each other (e.g., NAMEX to $NAMEX or NAMEX to NAMEX(2)). Those similar names usually show the existence of different encapsulated entities with the same content or assume the same content but having (small) differences. This situation leads to confusion about whether such an entity is valid or superfluous.

Measurement: Check whether identical encapsulated entities exist in the part.

Supporting information: Identical encapsulated entities may result from merging two parts or copying entities from one part to another.

Recommendation: Identical encapsulated entities must not exist in the part.

3.2.1.16 Empty encapsulated entities present: O-CM-EP

Problem description: An empty encapsulated entity does not contain any element but might be referenced in the CAD part. It has no addedvalue.

Measurement: Check whether empty encapsulated entities exist in the part.

Supporting information:

Recommendation: Delete empty encapsulated entities.

3.2.1.17 External item reference: O-CM-EI

Problem description: Most CAD systems are able to load external geometrical forms into a current part by the way of geometric references, not by duplication. This allows a large volume of data to be stored in the current part but requires the availability of those references, e.g., in the PDM system or regarding directories and part names. External references might cause problems such as unknown links or unclear paths, after data transfer.

Measurement: Check whether external item references are used.

Supporting information: The use of external references within a corporate organisation is useful because existing geometry is shared. These links, however, should deserve severe attention during removal, deletion, renaming or data exchanges.

Recommendation: If it is agreed to use external references, ensure transmission of referenced items during data exchange. Otherwise, do not use external references or replace them by the original geometry.

3.2.1.18 Inconsistent item reference: O-CM-IR

Problem description: Related to criteria 3.2.1.17, it is basically important to ensure the consistency of external item references.

Measurement: Check whether item references are consistent, i.e., each reference can be resolved.

Supporting information: See criteria 3.2.1.17.

Recommendation: All item references must be able to be resolved.

3.2.1.19 Non-standard simplified part: O-CM-SP

Problem description: In some process steps, simplified parts (with a simplified product shape) are used (e.g., DMU, viewing, etc.). Simplified parts with parameters (e.g. approximation accuracy) not according to the company standards might lead to confusion, incorrect measurements and during the transfer of those parts.

Measurement: Check whether a simplified representation exists in the part and whether its parameters are compliant with the company standard.

Supporting information: A simplified representation might be used to exchange geometry without its “know-how”, e.g., parametric.

Recommendation: The parameters for simplifying (e.g., approximation accuracy) must be compliant to the company standard.

3.2.1.20 Element outside bounding box: O-CM-OB

Problem description: Sometimes some elements are created outside the bounding box because of manipulation error or approximation error, etc. This may lead to problems in data exchange and to problems with batch applications (e.g., computation of mass properties and bounding box).

Measurement: Check whether there are elements outside the bounding box.

Supporting information:

Recommendation: Delete or do not use elements outside the bounding box.

3.2.2 Group / Layer

3.2.2.1 Group used: O-GL-GU

Problem description: If grouped data are loaded into a CAD system that does not support a grouping function, the process of sorting the data could demand extra workload or cause a shape or structure recognition error to occur.

Measurement: Whether a grouping function is used.

Supporting information:

Recommendation: If subject to agreement between the parties concerned, this function shall apply.

3.2.2.2 Number of groups exceeded: O-GL-NG

Problem description: Groups are used to organise geometric elements describing a part. Usual operations (like visualise, hide, move, or copy) of those groups are much easier than with single elements.

If grouped data are loaded into an application (e.g., CAE pre-processor) that is not able to read that grouping entity, this might lead either to a loss of information or extra workload to create a new structure.

For some CAD applications (e.g., kinematics), groups are basic requirements. Deletion of those groups might lead to the loss of important information for those applications.

Measurement: Check the number of groups.

Supporting information: A large number of groups might slow down the handling of the part.

Recommendation: Use a number of groups compliant to the company standard.

3.2.2.3 The same element registered with more than one group: O-GL-IG

Problem description: Having the same element registered with more than one group might create problems when selecting elements, might mesh a shape, and could lead to duplicated elements during data conversion between different CAD systems.

Measurement: Check whether the same element is not registered with more than one group.

Supporting information:

Recommendation: Do not register the same element in more than one group.

3.2.2.4 Non-standard grouping of elements: O-GL-IE

Problem description: Some companies recommend the standardised organisation of specific geometric elements in defined groups. A non-standard group is a group not including all expected elements or including more than the expected.

Forwarding parts with non-standard grouping might lead to confusion and time expense to understand and correct the data structure.

Measurement: Check whether elements are organised in groups according to the company standard.

Supporting information: In most cases, checking these criteria automatically is impossible and needs to be done manually.

Recommendation: Use grouping compliant to the company standard.

3.2.2.5 Non-standard group name: O-GL-GN

Problem description: Most companies, especially OEMs, have fixed specific naming conventions with the groups to ensure information quality and usability.

Measurement: Check whether the naming of a group is according to the company standard.

Supporting information: The length of the group name shall be limited to 31 characters or fewer. This is the recommendation based on the group name specifications for major CAD systems.

Recommendation: Use group naming compliant to the company standard.

3.2.2.6 Layer used: O-GL-LY

Problem description: If layered data are loaded into a CAD system that does not support a layer function, the process of sorting the data could demand extra workload or cause a shape or structure recognition error to occur.

Measurement: Whether two or more layer settings are used.

Supporting information: Because not all CAD systems come complete with a layer function, problems could arise in the course of data flow. Use of a layer function, therefore, should be agreed upon between the parties concerned beforehand.

Recommendation: If subject to agreement between the parties concerned, this function shall apply.

3.2.2.7 Number of layers exceeded: O-GL-NL

Problem description: Layers are used to organise geometric elements describing a part. Usual operations (like visualise, hide, move, or copy) of elements on layers are much easier than with single elements.

If data organised with layers are loaded into an application (e.g., CAE pre-processor) that is not able to read layers, this might lead either to a loss of information or extra workload to create a new structure.

Measurement: Check the number of layers used.

Supporting information: Number of layers shall be limited to 254 layers or fewer. This is the recommendation based on the maximum number of layers specifications for major CAD systems.

Recommendation: Use a number of layers compliant to the company standard

3.2.2.8 Wrong layer distribution of instances: O-GL-WL

Problem description: Instances can have their own layer distribution. It’s not reasonable to transfer it to the part to avoid decrease of manageability.

Measurement: Check layer distribution of instances.

Supporting information: An eventual applied layer structure in the instance transferred in the part can make it difficult to manage.

Recommendation: Use only transferred instances as one entire entity in the part.

3.2.2.9 Non-standard layer usage: O-GL-LU

Problem description: Each part consists of several layers, which can be considered as transparencies that can be displayed separately on the screen. A number or a name identifies each layer. To guarantee overview and handling, using special standards, companies only admit certain elements on certain layers.

In most systems, each element of the part is associated with one of these layers. There is always a current layer. Any new element is automatically created in the current layer. The current layer is always displayed on the screen.

Measurement: Check whether elements are registered with each layer according to predefined rules, and whether invalid elements are not registered.

Supporting information: If the layer structure is incorrect, this can lead to severe problems in further processes, e.g., for manufacturing or quality verification the wrong elements are plotted or displayed. Some data exchange procedures are using the layer distribution.

An eventual applied layer structure in the instance transferred in the part can make it difficult to manage.

Details may have detail data of other parts, and in this case, the layer rule of the other parts may be applied. Therefore, the layer information of the detail data must be checked.

Recommendation: Use layers compliant to the company standard.

3.2.2.10 Non-standard layer name: O-GL-LN

Problem description: Most companies, especially OEMs, have fixed specific naming conventions for layers to ensure information quality and usability. Layers are usually identified by name or number. Some CAD systems, however, accept layer names with alphabetic characters, while others allow layer numbers only in a specified range. But values in the range recommended above would be acceptable to any CAD system.

Measurement: Check whether the naming of a layer is according to the company standard.

Supporting information: A single-byte numeric numbering (e.g., 1 to 254) is usually used for naming.

Recommendation: Use layer naming compliant to the company standard.

3.2.2.11 Layer group used: O-GL-GL

Problem description: Layer groups are used to ease layer visualisation. If layer groups are transferred into an application (e.g., CAE pre-processor) that is not able to read that entity, this might lead either to a loss of information or extra workload to create a new structure.

Measurement: Check whether layer grouping is used.

Supporting information:

Recommendation: Use layer grouping compliant to the company standard.

3.2.2.12 Empty layer group: O-GL-EL

Problem description: An empty layer group does not contain any layer with geometry. It has no addedvalue.

Measurement: Check whether empty layer groups exist in the part.

Supporting information:

Recommendation: Delete empty layer groups.

3.2.2.13 Non-standard layer group: O-GL-LA

Problem description: Some companies recommend the standardised organisation of specific layers in defined layer groups. A non-standard layer group is a layer group not including all expected layers or including more than the expected.

Storing a part with the wrong current layer group can lead to visualisation confusion when opened by another person. Furthermore, certain conversion processors will only process active layers and layer groups.

Measurement: Check whether layers are organised in layer groups according to the company standard.

Supporting information: Layer groups allow easy switching from one product view to another.

Recommendation: Use layer grouping compliant to the company standard.

3.2.3 Co-ordinate systems

3.2.3.1 Local co-ordinate system used: O-CS-LS

Problem description: If the interface function does not support local co-ordinate systems, parts would be reproduced at a location different from the intent of the design, resulting in confusion.

Measurement: Check whether a local co-ordinate system exists in the part space.

Supporting information: While most major CAD systems support the setting of local co-ordinate systems, it may happen that the interface function does not support local co-ordinate systems.

Recommendation: Use local co-ordinate systems compliant to the company standard.

3.2.3.2 Non-reference co-ordinate system active: O-CS-NR

Problem description: Geometry creation in a CAD system is always related to a co-ordinate system. To ease data creation with specific translation and rotation parameters, it is possible to create auxiliary, non-reference co-ordinate systems. If dimension measurement or a similar operation is carried out without realizing that a non-reference co-ordinate system is active¸ this might lead to wrong dimension value or an invalid machining condition.

Measurement: Check whether the reference co-ordinate system is active.

Supporting information: The creation and analysis of elements refer to the co-ordinate system, which means that co-ordinates and angles are calculated in respect to the active co-ordinate system.

Recommendation: Ensure that the reference co-ordinate system is active.

3.2.3.3 Non-standard co-ordinate system orientation: O-CS-NO

Problem description: A wrong co-ordinate system orientation can lead to several problems in the process chain, e.g., reversed receiving geometry during data exchange or a reversed result when manufacturing (milling) the part.

Measurement: Check whether the co-ordinate system is a left- or right-hand oriented system.

Supporting information: Basically, there are two types of 3D axis systems: direct (right hand-oriented) and reversed (left hand-oriented).

An inverted axis system represents the reflection of a part, while the optical look remains equally. An inverted co-ordinate system has one direction changed from positive to negative.

Recommendation: Use orientation of co-ordinate systems compliant to the company standard. Do not intermix left- and right-hand systems.

3.2.3.4 Non-standard co-ordinate system name: O-CS-CN

Problem description: Several co-ordinate systems can exist simultaneously in one part. Therefore, the naming is important to identify the reference co-ordinate system and its change will lead to confusion and errors.

Measurement: Check the naming of the co-ordinate systems according to the company standards.

Supporting information:

Recommendation: Use the standard co-ordinate systems' name defined by the company standard.

3.2.3.5 Non-standard unit: O-CS-SU

Problem description: The part units describe the unit system used by the part. Units are the basis for any geometrical and physical computing. Using a Non-standard unit can lead to errors while evaluating the shape, doing analysis, or setting machining conditions.

Measurement: Check whether a part unit system complies with the company standard.

Supporting information: The mixed use of different unit systems (e.g., in assemblies) may cause a fatal error.

Recommendation: Use the standard unit defined in the company standard.

3.2.3.6 Non-standard scale: O-CS-SS

Problem description: Using a non-standard scale might lead to an error while evaluating the shape, doing analysis, or setting machining conditions.

Measurement: Check whether the part scale setting complies with the company standard.

Supporting information: Once a part scale is modified, one cannot determine whether it is the original part or has later been modified. Hence, part scale 1.0 is assumed to be standard, except in situations where multiplication by a shrinkage percentage is involved.

Recommendation: The standard part scale should be 1.0. If it is to be modified, current scale must be documented.

3.2.3.7 Transformation stored: O-CS-TS

Problem description: If a set of transfer information exists, a design error could occur by using a “wrong” transformation set.

Measurement: Check whether a transformation is stored in the part.

Supporting information:

Recommendation: Delete all transformations before data exchange.

3.2.4 Assembly

3.2.4.1 Assembly relationship used: O-AR-AR

Problem description: Some CAD systems use assembly relationships, which means the assembling (including translation and rotation) of parts to an assembly. The information about that assembly might be lost during data exchange or the receiving system might not be able to read this information, so that a receiver is not able to “reassemble” the product.

CAE analysis are usually carried out on a single item basis so that the presence or absence of an assembly representation should be defined beforehand.

Measurement: Check whether an assembly representation is used.

Supporting information:

Recommendation: Refer to the assembly usage recommendation defined in the company standard.

3.2.4.2 Undefined assembly constraints : O-AR-UC

Problem description: The positioning of a component in an assembly consists of suppressing its degrees of freedom (3 translations, 3 rotations) using assembly constrains, e.g., alignment. If the constraints are not defined, the problem will be the inability to reproduce the correct product shape and set correct analysis conditions.

A non-constrained assembly is liable to calculation problems in a subsequent process chain (such as kinematics and simulation).

Measurement: Check whether assembly constraints suppress the six degrees of freedom (translation in the x, y, and z-axis directions and rotation on each axis).

Supporting information: An under-constrained assembly may lead to computing problems in the data life cycle, e.g., kinematics’ simulation.

Recommendation: Design a fully constrained assembly.

3.2.5 Solid

3.2.5.1 History not used: O-SO-HN

Problem description: The use of a history largely affects the form change workload.

Implementing form changes to a model that does not have a history entails an enormous workload.

Measurement: Check whether a history is used

Supporting information:

Recommendation: Use a history.

3.2.5.2 History not updated: O-SO-HU

Problem description: If the history of a model is left non-updated, subsequent operations (such as a Boolean operation) could fail.

Measurement: The history of a model must have been updated after its modification.

Supporting information: The reason the update is not done might be because the sender did not perform it or the update is impossible due to severe problems in the geometry.

When an update is required:

- Data must be loaded from an external source through a data transfer.
- Some command must be given from CAD during modeling to prompt an update.

Recommendation: Update the history of a model.

3.2.5.3 Missing solid construction history: O-SO-MH

Problem description: Solid has incomplete model history — e.g., modelling features, parameters, or operations used (to define the solid) are not available.

Measurement: Whether complete construction history is available (system specific: e.g., via existence of Import Feature, Unparametrised Feature, etc.).

Supporting information: Often a history is required for later modifications on models (“History of origin”). In other cases, the history is intentionally deleted in order to protect engineering and modelling knowledge. In other instances, history is not required at all, e.g., mock-up.

Recommendation: Recreating a “History of origin” is often not possible. Therefore, avoid deletion or loss of a history in the original CAD model.

3.2.5.4 Unused solid construction history: O-SO-UH

Problem description: Some modelling features, parameters, or operations are defined but do not contribute to the solid (e.g., "dead branch"). This is often excess history information, not needed to support the model.

Measurement: Whether all construction history is used by the solid.

Supporting information: Excess information makes the model more difficult to understand and may lead to problems during update.

Recommendation: Clear the model of all elements that are not to be used.

3.2.6 Form features

3.2.6.1 Unresolved feature used: O-FE-UF

Problem description: Unresolved features are part elements which are not processed correctly, mostly due to a change of the underlying part geometry. The problem is the inability to represent the correct product shape.

Measurement: Check whether unresolved features exist in the current part.

Supporting information: Commonly this happens with fillets and chamfers. Unresolved features can lead to problems in the further process chain because they are not correctly defined. In many cases, this problem leads to non-updateable parts, but in some systems a part can be updated and still contain unresolved features.

Recommendation: Correct unresolved features.

3.2.6.2 Inactive feature used: O-FE-IF

Problem description: Inactive features or primitives are elements within the history tree temporarily not involved in any topological operation. The existence of inactive features might lead to confusion about its necessity and validity.

Measurement: Check whether inactive features exist in the current part.

Supporting information: Inactive features can be used for creating derivatives (variants) if one branch is representing variant A and another one variant B. If some part regions cannot be updated, errors might be bypassed by deactivating the branch (or the feature itself) containing the corrupt part geometry.

Recommendation: Avoid using inactive features.

3.2.7 Elements

3.2.7.1 Non-standard element name: O-EL-EN

Problem description: In some CAD systems, elements are named automatically, including continuous numbers. Gaps in the name sequence or duplicate names might confuse the receiver.

Measurement: Check whether the element names respect the company standard.

Supporting information: The standardised and consistent naming of elements used within a part is helpful for managing the data. For this purpose, many companies have defined naming conventions for certain part elements.

Recommendation: Change the name in order to make it consistent with the company standard.

3.2.7.2 Unused element present: O-EL-UE

Problem description: Elements that do not contribute to the product shape (unused elements) reduce the clarity of the description and might lead to problems during the execution of modifications. Those elements unnecessarily increase the file size, with the consequence of bad CAD system performance.

Measurement: Check whether unused elements exist that do not have a direct contribution to the product shape.

Supporting information:

- Examples for unused elements in the Solids area are:
- Auxiliary geometry "without children," i.e., geometry that is not required for the development of solids,
- A body that will be completely taken up by a second one,
- A body that lies outside a second one and will therefore be subtracted from it,
- Not used, hidden design steps (e.g.," dead branches").

Recommendation: Remove unused (auxiliary) elements.

3.2.7.3 Prohibited element used: O-EL-PE

Problem description: To avoid problems using "critical" elements, companies have developed lists of prohibited elements that must not be used in a part. An example for such entities is an "old" entity, created in former CAD version (e.g. ,faceted solid). Keeping such entities can lead to operation errors or data conversion errors.

Measurement: Check whether prohibited elements (according to the company standard) are used.

Supporting information: A list of prohibited elements must be provided by the company requiring these criteria.

Recommendation: Avoid or replace prohibited elements.

3.2.7.4 User-defined element used: O-EL-UD

Problem description: In some CAD systems, users are allowed to generate customised entities. Such entities are usually unknown outside the creating CAD system, i.e., will be lost during data transfer.

Measurement: Check whether user-defined elements are used.

Supporting information: User-defined entities might be used in a local purpose but should be replaced by standard exchangeable elements before data exchange.

Recommendation: Avoid or replace user-defined elements.

3.2.8 Presentation

3.2.8.1 Non-standard colour settings: O-PR-CO

Problem description: Changing the colour settings will lead to wrong visualisation of the product shape and therefore might lead to confusion the user's side.

Measurement: Check the colour settings according to the company standard.

Supporting information: Usually colours are identified by RGB values and/or colour numbers. In most CAD systems, the setting of standard colours is a "global" parameter, not changeable for a single part.

Recommendation: Use the company standard colour settings.

3.2.8.2 Non-standard element colour: O-PR-EC

Problem description: Some companies use element colours to structure a part for visualisation. Ignoring or violating the standard element colours might lead to confusion.

Measurement: Check colour settings of elements according to the company standard.

Supporting information: The use of colours with CAD-specific significance, such as a background or highlight colour, might lead to "invisible" elements.

Recommendation: Use element colours according to the company standard.

3.2.8.3 Non-standard point marker symbol: O-PR-PT

Problem description: In the case of incorrect point marker symbol usage, points might be invisible and/or unselectable.

Measurement: Check point marker symbols.

Supporting information:

Recommendation: Use point market symbols that comply with company standards agreed upon with customers as appropriate.

3.2.8.4 Non-standard line type: O-PR-LT

Problem description: In the case of incorrect line type usage, the design purpose might be incorrectly interpreted.

Measurement: Check the line type of each line.

Supporting information:

Recommendation: Use line type that complies with company standards agreed upon with customers as appropriate.

3.2.8.5 Non-standard line width: O-PR-LW

Problem description: In the case of incorrect line width usage, the design purpose might be incorrectly interpreted.

Measurement: Check the line width of each line.

Supporting information:

Recommendation: Use line width that complies with company standards agreed upon with customers as appropriate

3.2.8.6 Non-standard element visibility: O-PR-VE

Problem description: Visibility or usability of elements might be switched to "inactive," "hidden," "non-selectable," etc. by the user. Some applications in the process chain only register and use "selectable" and "non-hidden" entities, i.e., ignore those elements mentioned previously.

Measurement: Check the visibility status of each element according to the company standard.

Supporting information: "Hidden" elements often do not contribute to the shape, increasing the file size of the part, and reducing the CAD system performance. The element visibility includes element display attributes.

Recommendation: Use or change element visibility settings according to the company standard.

3.2.8.7 Non-standard display mode: O-PR-DM

Problem description: A part is stored with the current display mode (e.g., wire frame, hidden line removal, shaded, etc.). Most companies have defined a standard display mode for storage. A display-intensive mode (shaded mode) might impact the CAD system performance in parting operations.

Measurement: Check whether the display mode is according to the company standard.

Supporting information: The user may have to perform a display mode conversion to ease shape recognition. Shaded mode can be used to manually detect major surface faults, e.g., gaps, missing surface patches.

Recommendation: Use and store display mode according to the company standard.

3.2.8.8 Element identifier display: O-PR-ED

Problem description: In most CAD systems, elements can have an identifier. Those identifiers (names) might be permanently visible with the element or not. A large number of visible identifiers might supersede the shape visualisation.

Measurement: Check whether each element identifier display setting is according to the company standard.

Supporting information: For a very few specific entities in the part (e.g., laws, etc.), it could be useful to display the identifier. Identifier display also can be used to communicate describing information such as thickness.

Recommendation: Use and store identifier display according to the company standard.

3.2.8.9 Screen refit not performed: O-PR-SR

Problem description: While parting a part, the zoom factor is changed permanently. Storing a part with a zoom factor not showing all elements might lead to confusion. Refitting the screen can also point out elements that were inadvertently off in space. If an element exists outside the current screen, it might be overlooked, resulting in a design error, analysis error, or machining error.

Measurement: Check whether the active screen shows completely every element.

Supporting information: In some CAD systems, this has to be checked/performed manually.

Recommendation: Perform screen refit.

3.2.9 Sketch

3.2.9.1 Wrong degree of detail in a sketch: O-SK-WD

Problem description: Here, Sketches are understood as a basic element for further Solid operations. In most systems, a user has the choice to put design details (e.g., corners, chamfers, etc.) into the sketch or add them afterward with the help of solid functions. In the different systems, the one or the other method is preferred to support changes and to avoid problems (regarding continuity, changeability, etc.).

Measurement: **C**heck the degree of detail (e.g., number and size of geometric elements) of sketches.

Supporting information: A sketch containing too many and small details (e.g., corners, chamfers, etc.) is called an over-detailed sketch. The definition of “many” depends on the part complexity. Further operations on over-detailed sketches (extrusion to solids, etc.) might lead to unnecessarily high degree geometry, complexity, or design errors.

Recommendation: Create sketch complexity related to the company standard.

3.2.9.2 Not fully constrained sketch: O-SK-NC

Problem description: A sketch should be fully constrained so that all degrees of freedom are defined. Sketch constraints fundamentally reflect the design intent. A missing constraint might lead to unpredictable results when using, positioning, or dimensioning an element.

Measurement: Check whether the sketch is fully constrained.

Supporting information: Constraints enable one to control the different properties of objects in a sketch, e.g., dimensional (positions, line length) or functional (parallelism, alignment, tangency, etc.).

Recommendation: Create only fully constrained sketches (neither over-constrained nor under-constrained).

3.3 Drawing Quality Criteria Descriptions

This chapter will correct quality criteria applied to 2D CAD representations (drawings), which are still often used in the automotive industry. The group decided to sum up the drawing-related geometric and non-geometric quality criteria in this single chapter.

Since “Views” are important elements to structure a drawing, some basic characteristics are mentioned here to ensure usability:

- Views can be scaled; the geometry itself, however, may not be scaled.
- Drawing elements shall be generated in the “view” in which they are depicted.
- Views with detail sections/enlargements must have the same source (zero point, reference point) as the original view.

The following criteria apply to traditional drawings generated through a CAD system.

3.3.1 Tiny elements: D-GE-TI

Problem description: Drawing elements that fall short of a particular measure can lead to invalid elements due to degeneration during the exchange of data into a system environment of lesser accuracy.

Measurement: Length of drawing element.

Supporting information: Elements that fall short of a particular size by particular geometrical operations can lead to invalid elements and thereby to gaps. See 3.1.1.10 Tiny curve or segment: G-CU-TI for further information.

Recommendation: Delete tiny elements. If dependent elements (dimensioning, hatching) are available, these should be isolated prior to the deletion.

3.3.2 Embedded elements: D-GE-EM

Problem description: During the generation of a drawing, identical elements can unintentionally arise (i.e., several lines of varying or equal length over one another) that unnecessarily enlarge the space requirements of the model. For example, identical elements, also called double elements, often hinder the automatic recognition of continuous curve paths.

Measurement: : Whether there is a geometrical drawing element completely embedded within another within the designated accuracy.

Supporting information: See 3.1.1.8 Embedded curves: G-CU-EM

Recommendation: Delete identical elements. As long as the elements are exactly identical they can be deleted without any problem. Where several elements of varying length lie above one another, then under certain circumstances it is more meaningful to discover the longest element and delete the shorter ones.

3.3.3 ISO conformable texts: D-OR-SC

Problem description: During the generation of texts and dimensioning, specific national characters and mutated vowels (e.g., German umlauts and “ß”) can lead to transfer problems. Text characters numbering more than 70 per line as well as multi-line texts can lead to losses during transfer and for that reason are to be avoided or a special agreement reached concerning the problem. Conformance to the International Standards Organisation texts is preferred**.**

Measurement: : Whether there are special characters used in texts, identifiers, or dimensions.

Supporting information: (Move following text from Problem description to here:) Conformance to the International Standards Organisation texts is preferred.

Recommendation: Specific national characters and vowel mutations are to be replaced (e.g., ä with ae; ß with ss). Texts with more than 70 characters per line shall be split up into several individual texts. Multi-line texts shall be replaced by several single-line texts.

3.3.4 CAD source notice: D-OR-SN

Problem description: Drawings (e.g., Plots) often contain no reference concerning their CAD source data record. Because of this, the feedback of changes (e.g., in the tool manufacture) to the 3D CAD model is made all the more difficult.

Measurement: : Whether there is CAD source note visible in the drawing.

Supporting information:

Recommendation: Drawings must include a CAD source reference (name of system, version, storage address 3D CAD model, part/drawing index etc.).

3.3.5 References on external databases and libraries: D-OR-ER

Problem description: During the exchange of geometry, symbols, etc. from external libraries, the visibility/usability of such by the recipient shall be ensured.

Measurement: : Whether there is a reference to an external library in the drawing.

Supporting information:

Recommendation: By using, for example, symbols, drawing frames, or standard parts from external databases, either:

- Give an explicit reference regarding the existence of such references and discuss the exchange of the library, or
- Ensure the complete inclusion and export of those library parts in the exchange file.

3.3.6 External 2D drawing present: D-OR-XD

Problem description: Most CAD systems allow the derivation of a 2D representation of a 3D product in a separate, in-one-direction, logically linked 2D drawing. The existence of 2D drawings needs to be declared in the 3D part to avoid unwanted results in such a drawing when updating the 3D. A change in the 3D without respecting the separate drawing might lead to problems and errors when exchanging the drawing.

Measurement: Check whether external 2D drawings exist with a reference to the 3D part.

Supporting information:

Recommendation: Use external 2D drawings according to the company standard.

3.3.7 2D/3D linkage not present: D-OR-DL

Problem description: In a master-model concept, the master part contains the original representation and all related representation (e.g., drawings, simplified representations) is linked to it. The associatively enables efficient and automated changes, but missing or broken associatively can lead to inconsistent data. In the case where 2D and 3D exist without such linkage, modifications made to either 2D or 3D alone would result in loss of consistency.

Measurement: Check whether 2D and 3D are linked (associated).

Supporting information: Today the use of 2D and 3D representations is still mixed in current design processes. However, 3D will increasingly be used for the future and 3D will be a main stream in the design environment. In this case, it will be required to establish a linkage that 3D is a master and 2D derives from it.

Recommendation: Link 2D to the 3D.

3.3.8 2D drawing not updated: D-OR-DU

Problem description: Some CAD systems do not automatically update drawing views when the corresponding 3D is changed. In that case drawing views may not represent the current part. This might lead to severe problems, e.g., the inability to determine whether the 2D drawing or the 3D shape is valid.

Measurement: Check whether 2D drawings are updated if associated to a 3D part.

Supporting information: It could be dangerous to automatically modify a part without taking associated parts into account. As soon as a user releases a change of the 3D, he is responsible for the update of all related parts.

Recommendation: Update 2D drawings.

3.3.9 Number of drawing sheets exceeded: D-OR-ND

Problem description: In some CADsystems, more than one drawing sheet can be used in one part. Some automatic processes might fail if there is more than one drawing sheet.

Measurement: Check whether the number of drawing sheets is according to the company standard.

Supporting information:

Recommendation: Use drawing sheets compliant to the company standard.

3.3.10 Missing plot frame points: D-OR-PF

Problem description: The draft points defining the diagonal of the plot-area are not defined. This will lead to a failure when plotting in batch mode.

Measurement: Check whether the plot point(s) exist(s) in the current part.

Supporting information: Some companies require running a specific CAD application, attach specific attributes to such points, or use a concept with one point and additional plotting format information to enable plotting in batch mode.

Recommendation: Insert the plot point(s) with the required adjectives to the required place according to the company standard.

3.3.11 Unlimited size of view frames: D-OR-VF

Problem description: View frames are used for limiting the display of a 3D part in a 2D-view. This allows blanking unnecessary/unwanted 3D design elements from the drawing and improves the overview. Views with "infinite frame" can, in some cases, negatively influence the data transfer (e.g., via IGES).

Measurement: Check whether unlimited view frames exist.

Supporting information: In some CADsystems, layer filters are used instead of view frames.

Recommendation: Use views with frames limited to be inside the drawing frame or plot format.

3.3.12 Empty drawing view: D-OR-EV

Problem description: An empty view is defined as a view that does not contain any geometry except the compulsory axis system. Such a view does unnecessarily occupy (disc) space and might lead to confusion. A person checking the success of a data exchange is unable to determine whether a view has been originally empty or empty because of a data conversion error.

Measurement: Check whether the drawing contains a view that does not contain elements except the axis system.

Supporting information: In some particular cases it might be useful to have empty views for further processes (e.g., to apply information).

Recommendation: Remove empty views.

3.3.13 Non-standard view name: D-OR-VN

Problem description: Most companies, especially OEMs, have fixed, specific view-naming conventions in CAD guidelines to simplify the readability of a drawing. Not meeting these conventions might lead to confusion.

Measurement: Check the naming of the view according to the company standards.

Supporting information: An example of a view name is cut A-A, A-A or view X.

Recommendation: Replace the view name by a name consistent with the company standard.

3.3.14 More than one 2D co-ordinate system present: D-OR-CS

Problem description: In creating 2D geometric elements from a 3D part, some CAD systems create one co-ordinate axis that is represented by the original 3D co-ordinate axis. An additional 2D co-ordinate axis is required to carry out some operations (such as move). If a number of ambiguously defined 2D co-ordinate systems exist in one view at the same time, the original 2D co-ordinate axis would be difficult to identify.

Measurement: Check whether two or more 2D co-ordinate axes are present for one view.

Supporting information:

Recommendation: Retain only the reference 2D co-ordinate axis used when the view was created. Delete all other 2D co-ordinate axes.

3.3.15 Fake dimensions: D-OR-FD

Problem description: Drawings are usually generated from the 3D part using projection functionality which automatically extracts the value of the dimensions. Those values may be modified manually. That leads to inconsistencies between the graphical representation and the associated dimensions. Errors may occur in interpreting drawings and manufacturing parts in wrong dimensions.

Measurement: Check whether the current value of dimension of an element and the dimension noted by its dimension line are consistent within a designated accuracy.

Supporting information: In most CAD systems the fake dimension is marked automatically by the system.

Recommendation: Do not use fake dimensions. If they are used, search for fake dimensions and automatically update the value of the dimension(s).

3.3.16 Non-standard display accuracy of dimensions: D-OR-DI

Problem description: Drawing dimension values can be rounded off in a non-standard way and therefore differ from the 3D part. Depending on the deviation, this can cause misunderstandings and errors.

Measurement: Check the displayed dimensions according to company standards.

Supporting information: Appropriate precisions depend, e.g., on manufacturing and measuring tolerances

Recommendation: Ensure precision according to the company standard.

3.3.17 Associative dimension not present: D-OR-AD

Problem description: Drawing dimensions should always refer to the 3D master model. Non-associative dimensions result when the associatively to the 3D model is lost. This can lead to severe errors as changes in the 3D model are not reflected in the drawing.

Measurement: Check if dimensions are associative to the 3D model.

Supporting information:

Recommendation: Make all dimensions associative.

3.3.18 Non-standard view dependent object: D-OR-VD

Problem description: In some CAD systems it is possible to generate additional objects (line, points) in an associative view or to modify objects (e.g., visibility) only in one view. This might lead to confusion.

Measurement: Check whether there are view dependent objects.

Supporting information:

Recommendation: Use view-dependent objects compliant to the company standard.

3.3.19 Wrong view projection method: D-OR-VP

Problem description: The different companies/countries use different methods to create standard, 2D views of a part in a drawing (U.S. vs. European projection method, 1st vs. 3rd Angle projection, etc.). A wrong projection method may lead to a misinterpretation of a drawing and failures in manufacturing.

Measurement: Check whether the projection method is according to the company standard.

Supporting information: Since a projection method used is not directly visible for a view, an interpreter relies on the method announced close to the drawing title block. Ensure that all views cope with that method.

Recommendation: Use a projection method compliant to the company standard.

4 CAE Data

During the products development phase, CAE data are created between various milestones. Therefore, the transfer of mesh data (made by pre-processors (meshers)) has increased between partners. This will keep expanding in the future. The purpose of this chapter is to describe the product data quality criteria associated with CAE data.

To compute the physical behaviour of a part, a device, an accessory or a product, the partial derivative equations which are concerned are to be solved. Several methods may be used to do so :

- finite differences method
- finite volumes method
- boundary elements method
- finite element method

The finite element method is the most important and the most frequently used method for performing these simulations. The physical analysis involved includes thermal, static, and dynamic elasticity simulations (linear or non linear), crash simulations, acoustic or electromagnetic simulations, fluid simulations, etc … For the finite element method, the mesh is the main part of the data model. Most of the quality criteria for the mesh that are defined hereafter may be uses for every purpose, but some criteria are relative to a specific type of analysis.

The 2D analysis is not treated in the present version of this guideline. Only the 3D analysis is presented.

Data that have to be prepared before the computation have several subsets, and the mesh is the most expensive part of them. That is to say, nothing is said here about the following:

- physical properties,
- limit conditions or initial conditions,
- solicitations : forces, temperature, initial velocities, initial stresses …

The mesh has two purposes :

- to define the geometry for the simulation
- to control the level of the approximation of the computed solution

The mesh is not independent of the type of simulation that will be done, but on the quality of the mesh point of view some general concepts may be proposed.

Several types of finite elements may be used to perform the physical simulation requested by the user. The simplest types of finite elements arealso the most frequently used. They are the following:

1) TRIA3, with 3 nodes and 3 linear edges where :

- $x(u,v) = x_1.f_1(u,v) + x_2.f_2(u,v) + x_3.f_3(u,v)$ $f_1(u,v) = 1\text{-}u\text{-}v$ $u = 0$
- $y(u,v) = y_1.f_1(u,v) + y_2.f_2(u,v) + y_3.f_3(u,v)$ $f_2(u,v) = u$ $v = 0$
- $z(u,v) = z_1.f_1(u,v) + z_2.f_2(u,v) + z_3.f_3(u,v)$ $f_3(u,v) = v$ $u+v = 1$
- $F(u,v) = F_1.f_1(u,v) + F_2.f_2(u,v) + F_3.f_3(u,v)$

Here x_1,y_1,z_1 are the co-ordinates of the node 1 and likewise for the nodes 2 and 3. F(u,v) is, inside this triangle, the field of the physical value that is to be computed. It may be a temperature field, a displacement (dx,dy,dz) field, or any other field according to the simulations that are to be done. F_1, F_2 and F_3 are the values of this field at each node of this triangle; they will be known with the results of the solving phase of the simulation.

2) isoTRIA6, with 6 nodes and 3 parabolic edges where :

- $x(u,v) = x_1.s_1(u,v) + ... + x_6.s_6(u,v)$ $s_1(u,v) = (1-u-v)(1-2u-2v)$
- $y(u,v) = y_1.s_1(u,v) + ... + y_6.s_6(u,v)$ $s_2(u,v) = u(2u-1)$
- $z(u,v) = z_1.s_1(u,v) + ... + z_6.s_6(u,v)$ $s_3(u,v) = v(2v-1)$ $u=0$
- $F(u,v) = F_1.s_1(u,v) + ... + F_6.s_6(u,v)$ $s_4(u,v) = 4u(1-u-v)$ $v=0$

$s_5(u,v) = 4uv$ $u+v=1$

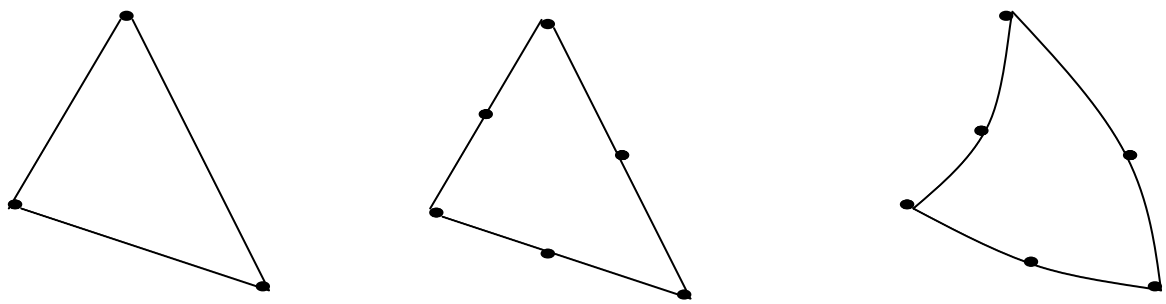
$s_6(u,v) = 4v(1-u-v)$

Here the field F(u,v) and the generic point of the triangle are defined by second degree polynomials of the two parameters u and v. TRIA3 and TRIA6 are called isoparametrical elements because they are used to define the geometry and the physical values of the same functions--here, the 2D polynomials of Lagrange.

3) linTRIA6, with 3 main nodes and 3 rectilinear edges with the 3 other nodes in the middles of the 3 edges, where :

- $x(u,v) = x_1.f_1(u,v) + x_2.f_2(u,v) + x_3.f_3(u,v) = x_1.s_1(u,v) + ... + x_6.s_6(u,v)$
- $y(u,v) = y_1.f_1(u,v) + y_2.f_2(u,v) + y_3.f_3(u,v) = y_1.s_1(u,v) + ... + y_6.s_6(u,v)$
- $z(u,v) = z_1.f_1(u,v) + z_2.f_2(u,v) + z_3.f_3(u,v) = z_1.s_1(u,v) + ... + z_6.s_6(u,v)$
- $F(u,v) = F_1.s_1(u,v) + F_2.s_2(u,v) + F_3.s_3(u,v) + F_4.s_4(u,v) + F_5.s_5(u,v) + F_6.s_6(u,v)$

Here only the field F(u,v) is defined by second degree polynomials of the two parameters u and v. Obviously linTRIA6 is not isoparametrical, and it has the same geometry as TRIA3.



TRIA3 linTRIA6 isoTRIA6

When there is no need to define the difference, TRIA6 means either a linTRIA6 or a isoTRIA6.

4) QUAD4, linQUAD8, isoQUAD8,and most simply QUAD8 are defined by the same manner on the base of a quadrilateral. For example, QUAD4 has 4 nodes Pij and may be defined by:

P(u,v) = $\sum_{i=1}^{i=2} \sum_{j=1}^{j=2} P_{ij} f_i(u) f_j(v)$ and F(u,v) = $\sum_{i=1}^{i=2} \sum_{j=1}^{j=2} F_{ij} f_i(u) f_j(v)$

-1 = u = 1 -1 = v = 1 $f_1(t) = (1 - t) / 2$ $f_2(t) = (1 + t) / 2$

To define a QUAD8 the best way is to begin with the QUAD9 which has 9 nodes and may be defined as the complete Lagrange element by:

P(u,v) = $\sum_{i=1}^{i=3} \sum_{j=1}^{j=3} P_{ij} f_i(u) f_j(v)$ and F(u,v) = $\sum_{i=1}^{i=3} \sum_{j=1}^{j=3} F_{ij} f_i(u) f_j(v)$

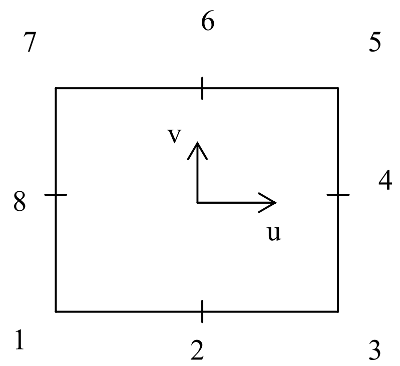
-1 = u = 1 -1 = v = 1 $f_1(t) = (t^2 - t) / 2$ $f_2(t) = 1 - t^2$ $f_3(t) = (t^2 + t) / 2$

As it is often not easy to use the central node P_{22}, it is better to define the isoQUAD8 as an incomplete Lagrange element that does not have a central node:

$f_i(u,v) = (-1+u_iu+v_iv)(1+u_iu)(1+v_iv) / 4$ i = 1, 3, 5, 7

$f_i(u,v) = (1-u^2)(1+v_iv) / 2$ i = 2, 6

$f_i(u,v) = (1+u_iu)(1-v^2) / 2$ i = 4, 8



linQUAD8 is an isoQUAD8 where the intermediate node of each edge is on the middle of this edge.

5) TETRA4, linTETRA10, isoTETRA10,and most simply TETRA10 are defined by the same manner on the base of a tetrahedron. For example, TETRA4 has 4 nodes and may be defined by:

$$P(u,v,w) = \begin{pmatrix} x(u,v,w) \\ y(u,v,w) \\ z(u,v,w) \end{pmatrix} = P_1 (1 - u - v - w) + P_2 u + P_3 v + P_4 w$$

$$F(u,v,w) = F_1 (1 - u - v - w) + F_2 u + F_3 v + F_4 w \qquad u+v+w=1 \quad u=0 \quad v=0 \quad w=0$$

6) PENTA6, linPENTA15, isoPENTA15, and most simply PENTA15 are defined by the same manner on the base of a pentahedron.

7) HEXA8, linHEXA20, isoHEXA20, and most simply HEXA20 are defined by the same manner on the base of a hexahedron. For example, HEXA8 has 8 nodes P_{ijk} and may be defined by:

$$P(u,v,w) = \begin{pmatrix} x(u,v,w) \\ y(u,v,w) \\ z(u,v,w) \end{pmatrix} = \sum_{i=1}^{i=2} \sum_{j=1}^{j=2} \sum_{k=1}^{k=2} P_{ijk} \; f_i(u) f_j(v) f_k(w)$$

and, of course: $$F(u,v,w) = \sum_{i=1}^{i=2} \sum_{j=1}^{j=2} \sum_{k=1}^{k=2} F_{ijk} \; f_i(u) f_j(v) f_k(w)$$

$$-1 = u = 1 \qquad -1 = v = 1 \qquad -1 = w = 1 \qquad f_1(t) = (1 - t) / 2 \quad f_2(t) = (1 + t) / 2$$

8) PYRAMID5, linPYRAMID13, isoPYRAMID13, and most simply PYRAMID13 are defined by the same manner on the base of a pyramid. For example, PYRAMID5 has 5 nodes (S, P_{11}, P_{12}, P_{21} and P_{22}) and may be defined by:

$$P(u,v,w) = \begin{pmatrix} x(u,v,w) \\ y(u,v,w) \\ z(u,v,w) \end{pmatrix} = w\,S + (1-w) \sum_{i=1}^{i=2} \sum_{j=1}^{j=2} P_{ij} \; f_i(u) f_j(v)$$

and, of course : $$F(u,v,w) = w\,F_S + (1-w) \sum_{i=1}^{i=2} \sum_{j=1}^{j=2} F_{ij} \; f_i(u) f_j(v)$$

$$w = 0 \quad w+u = 1 \quad w-u = 1 \quad w+v = 1 \quad w-v = 1 \quad f_1(t) = (1 - t) / 2 \quad f_2(t) = (1 + t) / 2$$

Some other types of finite elements may also be used. As seen here, the finite element has a geometrical aspect and a physical aspect. About the geometrical quality of the mesh, a finite element which is not in this list may be replaced by one of these.

It is important to note that a TETRA has 4 TRIA faces, a PENTA has 2 TRIA faces and 3 QUAD faces, a HEXA has 6 QUAD faces, and a PYRAMID has 1 QUAD face and 4 TRIA faces.

For any volumic finite element, the space where parameters u, v and w may be used is named the parametric space or the parent space. And the space inside the finite element itself is named the geometrical space or the child space. We have:

dV = dx dy dz = J(u,v,w) du dv dw where : $J(u,v,w) = \det \begin{pmatrix} \frac{.x}{.u} & \frac{.x}{.v} & \frac{.x}{.w} \\ \frac{.y}{.u} & \frac{.y}{.v} & \frac{.y}{.w} \\ \frac{.z}{.u} & \frac{.z}{.v} & \frac{.z}{.w} \end{pmatrix}$

So : $J_{moy} = V_{child} / V_{parent}$ For the TETRA4, P(u,v,w) is of first degree in relation to u, v and w. That is why in this element J = cte = J_{moy} . For other volumic elements J(u,v,w) is a polynomial which is often complex. And if J has a big range of variations over the space of the element, then this is the result of a big geometrical distortion of the element. As it is quite difficult and long to compute J_{max} and J_{min} all over the element, it is easier and sufficient to compute $J_i = J(u_i,v_i,w_i)$ only at each point (u_i,v_i,w_i) used in this element to approximate the computation of an integral over the element by a reduced integration formula:

$$I = \int_V G(x,y,z)dxdydz = \int_W H(u,v,w)dudvdw \approx \sum_{i=1}^{i=r} a_i H(u_i, v_i, w_i)$$

Where : H(u,v,w) = G(x(u,v,w),y(u,v,w),z(u,v,w)) J(u,v,w)

r is defined according to the degree of form functions of the element. The r values a_i and the r points (u_i,v_i,w_i) are defined in order to have this formula exact to as many polynomials as possible. They are known and given in tables. They are specific for each element. These points are sometimes named Gauss Legendre points.

For a TRIA element we have :

$$I = \int_0^1 \int_0^{1-u} H(u,v)dudv = \sum_{i=1}^{i=r} a_i H(u_i, v_i)$$ with:

r	u_i v_i	a_i
1	1/3 1/3	1/2
3	1/2 0	1/6
	1/2 1/2	1/6
	0 1/2	1/6
3	2/3 1/6	1/6
	1/6 1/6	1/6
	1/6 2/3	1/6
4	3/5 1/5	25/96
	1/5 1/5	25/96
	1/5 3/5	25/96
	1/3 1/3	-27/96

For a TETRA element we have:

$$I = \int_0^1 \int_0^{1-u} \int_0^{1-u-v} H(u,v,w)\,du\,dv\,dw = \sum_{i=1}^{i=r} a_i H(u_i, v_i, w_i) \qquad \text{with:}$$

r	u_i v_i w_i	a_i
1	1/4 1/4 1/4	1/6
4 a = (5 - $\sqrt{5}$) / 20 B = (5 + 3 $\sqrt{5}$) / 20	a a a a a b a b a b a a	1/24 1/24 1/24 1/24
5 a = 1/4 b = 1/6 c = 1/2	a a a b b b b b c b c b c b b	-2/15 3/40 3/40 3/40 3/40

For a QUAD element we have:

$$I = \int_{-1}^{+1}\int_{-1}^{+1} H(u,v)\,du\,dv = \sum_{i=1}^{i=r1}\sum_{j=1}^{j=r2} a_i b_j H(u_i, v_j)$$

And for a HEXA element we have:

$$I = \int_{-1}^{+1}\int_{-1}^{+1}\int_{-1}^{+1} H(u,v,w)\,du\,dv\,dw = \sum_{i=1}^{i=r1}\sum_{j=1}^{j=r2}\sum_{k=1}^{k=r3} a_i b_j c_k H(u_i, v_j, w_k) \qquad \text{with:}$$

r	u_i	a_i
1	0	2
2	-1/ $\sqrt{3}$ +1/ $\sqrt{3}$	1 1
3	- $\sqrt{3/5}$ 0 + $\sqrt{3/5}$	5/9 8/9 5/9
4 a = $\sqrt{\left(3+2\sqrt{6/5}\right)/7}$ b = $\sqrt{\left(3-2\sqrt{6/5}\right)/7}$	-a = -0.86113... -b = -0.33998... +b = +0.33998... +a = +0.86113...	1/2 - 1/6 $\sqrt{6/5}$ = 0.34785... 1/2 + 1/6 $\sqrt{6/5}$ = 0.65214... 1/2 + 1/6 $\sqrt{6/5}$ = 0.65214... 1/2 - 1/6 $\sqrt{6/5}$ = 0.34785...

To qualify the quality of the mesh, some criteria will be defined. A mesh respecting these criteria will be said to have the SASIG-PDQ quality. These criteria are shown here:

Table 6. CAE Data Criteria

			Shell		*Solid*			
Criteria No.	***Item name***	*Reason*	**TR**IA **A-TR**	**QU**AD **A-QU**	**TE**TRA **A-TE**	**PE**NTA **A-PE**	**PY**RAMID **A-PY**	**HE**XA **A-HE**
4.1	**TI**ny finite element **TI**	performance, time step	A-TR-TI	A-QU-TI	A-TE-TI	A-PE-TI	A-PY-TI	A-HE-TI
4.2	**M**inimum **A**ngle of triangular element **MA**	element shape depending on accuracy	A-TR-MA		A-TE-MA	A-PE-MA	A-PY-MA	
4.3	**WA**rpness **WA**	difference from model		A-QU-WA		A-PE-WA	A-PY-WA	A-HE-WA
4.4	**SK**ew angle **SK**	element shape depending on accuracy		A-QU-SK		A-PE-SK	A-PY-SK	A-HE-SK
4.5	**TA**per **TA**	element shape depending on accuracy		A-QU-TA		A-PE-TA	A-PY-TA	A-HE-TA
4.6	**AS**pect ratio **AS**	element shape depending on accuracy	A-TR-AS	A-QU-AS	A-TE-AS	A-PE-AS	A-PY-AS	A-HE-AS
4.7	**FR**ee face **FR**	miss modeling			A-TE-FR	A-PE-FR	A-PY-FR	A-HE-FR
4.8	**CO**ntinuity **CO**	element shape depending on accuracy	A-TR-CO	A-QU-CO				
4.9	**ST**retch **ST**	element shape depending on accuracy			A-TE-ST			
4.10	**S**ize of the **M**odel **SM**	element shape depending on accuracy	A-TR-SM	A-QU-SM	A-TE-SM	A-PE-SM	A-PY-SM	A-HE-SM
4.11	**JA**cobian **JA**	element shape depending on accuracy			A-TE-JA	A-PE-JA	A-PY-JA	A-HE-JA
4.12	Middle **P**oint **D**eviation **PD**	element shape depending on accuracy	A-TR-PD	A-QU-PD	A-TE-PD	A-PE-PD	A-PY-PD	A-HE-PD
4.13	Middle **P**oint **A**lignment **PA**	element shape depending on accuracy	A-TR-PA	A-QU-PA	A-TE-PA	A-PE-PA	A-PY-PA	A-HE-PA

4.1 Tiny finite element: A-TR-TI A-QU-TI A-TE-TI A-PE-TI A-PY-TI A-HE-TI

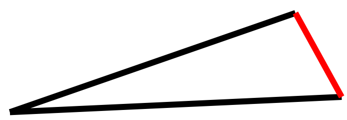
Problem description: During the meshing process, the software packages use automatic algorithms upon some rules. Resulting mesh may contain small elements. These involve an increase of the time of some computations.

Concerned elements: All types

Measurement : The length of the smallest edge of the finite element.

Supporting information: Usually finite element simulations have no difficulties with small elements, but the crash analysis often uses an explicit method to iterate time steps. In this method the time step has to be smaller than the time used by sound to go from one vertex to an other. That is why, in this case, all the small elements have to be greater than one limit given by the user.

Recommendation: Rebuild the mesh in order to avoid tiny finite elements



Example : Tiny finite element

4.2 Minimum angle of triangular element: A-TR-MA A-TE-MA A-PE-MA A-PY-MA

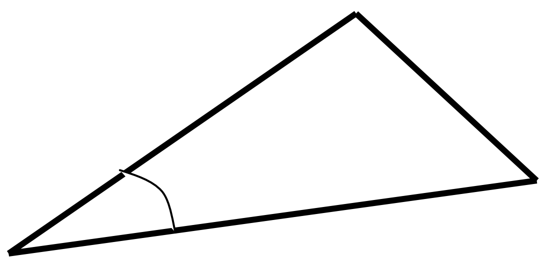
Problem description: Any angle of a triangular element has to be greater than a minimum value.

Concerned elements: All TRIA elements and TRIA faces of TETRA, PENTA and PYRAMID elements.

Measurement: Angles of the triangle.

Supporting information: A too-small angle suggests a near-degenerate element.

Recommendation: Use a better rule to build the mesh or rebuild the mesh locally.



Example : Minimum angle of a triangular element or triangular face

4.3 Warpness: A-QU-WA A-PE-WA A-PY-WA A-HE-WA

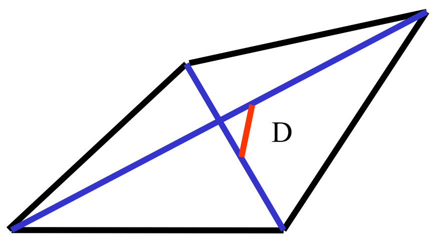
Problem description: The warpness of the quadrilateral means that the geometry is not well represented.

Concerned elements: All QUAD elements and QUAD faces of PENTA, HEXA, and PYRAMID elements.

Measurement: Ratio of distance between the diagonals by the maximum edge length.

Supporting information: A toolarge warpness ratio suggests that the mesh is too far from the geometry.

Recommendation: Rebuild the mesh locally using smaller elements.



Example : Warpness of a QUAD element or quadrilateral face

4.4 Skew angle: A-QU-SK A-PE-SK A-PY-SK A-HE-SK

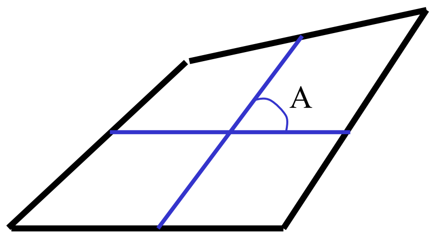
Problem description: The effect of a skew angle is like a lozenge that is folded on itself.

Concerned elements: all QUAD elements and QUAD faces of PENTA, HEXA and PYRAMID elements.

Measurement: S = (90° - A) where A is the angle in degrees between the two lines joining the opposite middles of the QUAD. If they are not on a plane, take a parallel line of one line passing by a point of the other line.

Supporting information: A too-small skew angle suggests a near-degenerate element.

Recommendation: Rebuild the mesh locally.



Example : Skew angle of a QUAD element or quadrilateral face

4.5 Taper: A-QU-TA A-PE-TA A-PY-TA A-HE-TA

Problem description: The effect of a taper is like a trapezoid close from a triangle.

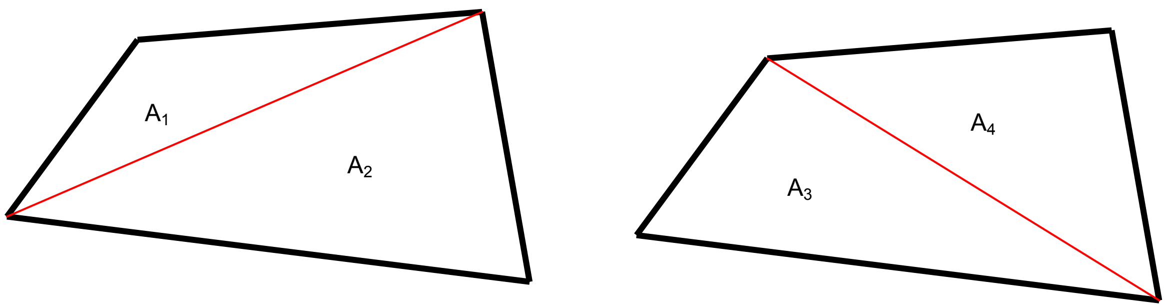
Concerned elements: All QUAD elements and QUAD faces of PENTA, HEXA, and PYRAMID elements.

Measurement: Divide the QUAD element into two triangles using the first diagonal, and then the same with the second diagonal. Compute all four areas A_i .

Compute $A_m = 0.25(A_1+A_2+A_3+A_4)$ and $Q = \underset{i}{Max} \mid A_i - A_m \mid / A_m$

Supporting information: Q = 0 for a rectangle. Q > 0.5 may be seen as a bad value.

Recommendation: Rebuild the mesh locally.



Example : Taper

4.6 Aspect Ratio: A-TR-AS A-QU-AS A-TE-AS A-PE-AS A-PY-AS A-HE-AS

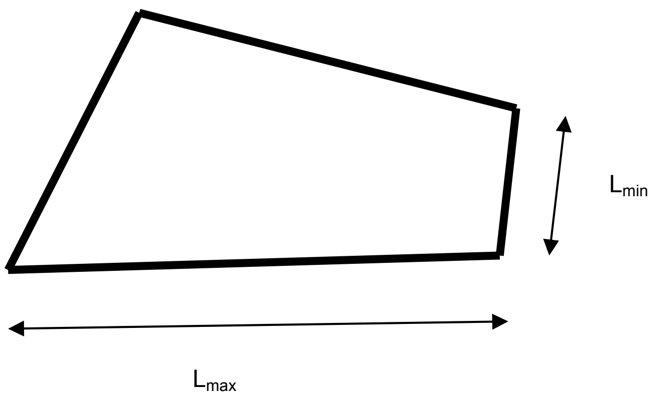
Problem description: When a finite element has two vertices that are too clos,e it is quite a degenerate element. This may lead to a bad conditioned system at the solving phase.

Concerned elements: All types.

Measurement: Ratio = L_{min} / L_{max} where L_{min} is the minimum length and L_{max} the maximum length of the edges of the element.

Supporting information: An edge must not be too small compared to the length of the element.

Recommendation: Rebuild the mesh to avoid tiny relative edges of finite elements.



Example : Aspect Ratio

4.7 Free faces: A-TE-FR A-PE-FR A-PY-FR A-HE-FR

Problem description: A face usually belongs to two elements: one for each side. Free faces are the faces belonging to only one element. Usually there is an outer boundary – the skin of the model - made of free faces, but inside the model a free face is a mistake coming from:

- an element or several elements that are forgotten
- inconsistent elements

Concerned elements: All types of volumic finite elements: TETRA, PENTA, PYRAMID, or HEXA.

Measurement: A free face belongs to only one element, but if it belongs to the outer boundary it is not a mistake.

Supporting information: Inconsistent elements involve errors of computation, but in some cases with particular additional conditions inconsistent elements are allowed. For example, it may be the case for an acoustical and vibrations analysis: the mesh of the solid having vibrations is thin but the mesh of the surrounding fluid for acoustic waves is coarse, so one acoustic element is in front of several mechanic elements. In this case there are also the corresponding coupled equations. It may be said that it is also the case when beams are coupled with classical mechanical finite elements.

Recommendation: Check manually, whether such free face is on the skin or not.

4.8 Continuity: A-TR-CO A-QU-CO

Problem description: In a mesh, a finite element and any next finite element need not be too different in size. This criterion is useful mainly for surfacic meshes.

Concerned elements: All types of surfacic finite elements: TRIA or QUAD.

Measurement: L_{max1} / L_{min2} where L_{max1} is the length of the maximum edge of this element and L_{min2} is the length of the minimum edge of the next element.

Supporting information: A surfacic element and one of its next elements have a common edge.

Recommendation: Correct the mistake.

4.9 Stretch: A-TE-ST

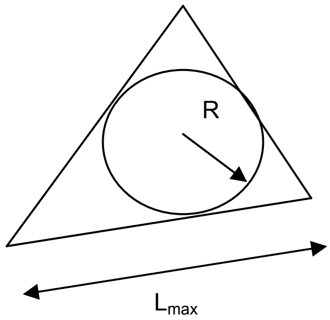
Problem description: Tetrahedric elements have to be stretched in order to be regular enough.

Concerned elements: TETRA4 and linTETRA10 elements

Measurement: $S = R / (L_{max} \sqrt{24})$ where R is the radius of the inscribed sphere and L_{max} is the length of the biggest edge.

Supporting information: S = 1 for a equilateral tetrahedron.

Recommendation: Rebuild the mesh locally.



Example : Stretch

4.10 Size of the model: A-TR-SM A-QU-SM A-TE-SM A-PE-SM A-PY-SM A-HE-SM

Problem description: In order to avoid too long solving times, it may be helpful to limit the number of nodes in the model.

Concerned elements: All elements.

Measurement: The number of nodes used by all sets of elements.

Supporting information: The limit may be specific to the tools used by the company to compute the solution.

Recommendation: Try to rebuild the mesh with coarser elements.

4.11 Jacobian: A-TE-JA A-PE-JA A-PY-JA A-HE-JA

Problem description: For any volumic element, a toolarge range of variations of the Jacobian means a toolarge geometrical distortion of the element.

Concerned elements: all types of volumic finite elements : TETRA, PENTA, PYRAMID, or HEXA.

Measurement: Ratio = J_{max} / J_{min} where $J(u_i,v_i,w_i)$ for i = 1 to r is evaluated at the r points used by the technology of the element to compute integrals by the reduced integration method.

Supporting information: A TETRA4 has always $J_{max} = J_{min}$ and so Ratio = 1

Recommendation: Rebuild the mesh locally.

4.12 Middle point deviation: A-TR-PD A-QU-PD A-TE-PD A-PE-PD A-PY-PD A-HE-PD

Problem description: When an edge is defined by three nodes, the intermediate node should not be too far from the middle point between the first and the third nodes.

Concerned elements: isoTRIA6, isoQUAD8, isoTETRA10, isoPENTA15,

isoPYRAMID13, and isoHEXA20

Measurement: Ratio = D / L where D is the distance from the intermediate node to the line going from the first node to the third node and L is the distance from the first node to the third node.

Supporting information:

Recommendation: Rebuild the mesh locally.

4.13 Middle point alignment: A-TR-PA A-QU-PA A-TE-PA A-PE-PA A-PY-PA A-HE-PA

Problem description: When an edge is defined by three nodes, the intermediate node should not be too far from the middle between the first and the third node.

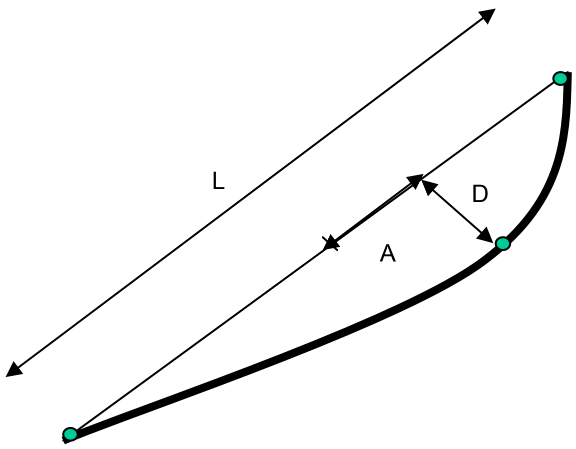
Concerned elements: isoTRIA6, isoQUAD8, isoTETRA10, isoPENTA15,

isoPYRAMID13, and isoHEXA20

Measurement: Ratio = A / L A is the distance between the projection of the intermediate node to the line going from the first node to the third node and the middle point between the first and the third node. L is the distance from the first node to the third node.

Supporting information:

Recommendation: Rebuild the mesh locally.



Example : Middle point deviation and middle point alignment

5 PDM Data

This section is intended to capture and present the data required and the quality of that data for Product Data Management applications and a robust optimised utilisation of product data. This would include many aspects of data management including tracking, recording, security, exchange, interoperability, archiving, and many other aspects of the product data life cycle. This section will be completed in a future version of this set of guidelines. The following are expected subsections, providing a sense of what will be covered.

- Bills of Material
- Configuration Management Data
- Associatively
- Product Structure (Assemblies)

6 Inspection Data

This section is intended to capture and present the data required and the quality of that data for inspection applications and processes. This would include many aspects of inspection data such as points, scans, exchange, and others. This section will be completed in a future version of this set of guidelines. The following are expected subsections, providing a sense of what will be covered.

6.1 CMM and other sensing devices

(Editor's **Note:** There are CMM data quality activities going on in the U.S., directed by AIAG and the National Institute of Standards and Technology (NIST), in which the auto industry is participating. It is expected that this work will address this section.)

6.2 Gauging

Under development.

7 Prototyping Data

This section is intended to capture and present the data required and the quality of that data for Rapid Prototyping. This would include many aspects of rapid prototyping including creation, manipulation, exchange, interoperability, repairing, and many other aspects of the rapid prototyping processes. This section will be completed in a future version of this set of guidelines. The following are expected subsections, providing a sense of what will be covered.

- Stereo Lithography (STL)
- Fusion Discharge Machining (FDM)
- Layer Object Modelling (LOM)

8 Manufacturing Data

This section is intended to capture and present the data required and the quality of that data for manufacturing applications and a robust optimised utilisation of the product data in a manufacturing environment. This would include many aspects of data in the manufacturing environment and processes. This section will be completed in a future version of this set of guidelines. The following are expected subsections, providing a sense of what will be covered.

- Tolerance
- Bills of Material
- CAM
- Virtual Assembly
- Process Data
- Jigs and Fixtures
- Tooling
- Manufacturing Process Analysis Tools
- Manufacturing Analysis Tools (e.g., mould flow analysis)

9 Quality Stamp

This section is intended to define an indicator for the CAD data quality described in chapters 3 to 8.

9.1 Background

During a products development phase, CAD data should be checked at various milestones with progressive check profiles and tools. In many cases the check result will display a number of violations against the checked quality criteria. At some of the above mentioned milestones CAD data has to be corrected to zero faults before further usage (e.g. final storage in the PDM system), but in other cases even bad quality data might be used.

The idea is to create one part of the check result documentation following standardised rules and store it inside or close to the CAD data to indicate it's effective quality – the quality stamp.

The standardisation of content and format of the quality stamp is essential for it's usability, e.g. to avoid confusion or misinterpretation at receivers site.

With the quality stamp the SASIG-PDQ working group also forces to use the common definition and encoding for the different quality criteria (see section II), e.g. G-SU-LG .

9.2 Example

The data exchange between Supplier and Customer is a classical example for such a milestone. Supplier checks the data before sending according to the rules given by the Customer.

This may take some minutes, in case of large data files up to hours. After receipt at the Customer, a new "incoming-check" usually takes place. This might take hours again. A quick transmission is obstructed by this.

Instead of double-checking the data, before sending and after receipt, a single check including creation of a "quality stamp" is able to reduce path-through-time to minimum.

9.3 Fundamental bilateral agreements before productive usage

Assuming to use different checktools at senders and receivers side, it should be assured that both checkresults are similar. A comparability benchmark between "conformance tested" checktools showed that a 100% equality is not given for all defined criteria because of different algorithms used. When using the same check tool in the same version, the comparability of checkresults is clearly more certain, nevertheless differences cannot finally be avoided because of different Operating systems or CAD systems. Sender and receiver should agree upon how to handle differences between check results.

Another important basic prerequisite to be agreed upon is a mechanism to avoid manipulation of the quality stamp. Especially, the modification of the CAD file or the quality stamp has to be prevented. For that purpose, a time stamp (date of storage) is designed. Content as well as realisation details still have to be defined in cooperation with the checktool providers.

The quality stamp consists of optional and mandatory information fields. Some optional fields might be declared as mandatory based upon bilateral agreement between sender and receiver.

With its receipt, the quality stamp itself will be checked regarding its consistence to the relevant CAD model (model changed after check?), regarding the consistence of used check profile to the requested one and regarding the existance of major (forbidden) errors. Sender and receiver should agree upon activities to start in case of major infringements (escalation steps).

9.4 Quality Stamp Content

Table 7. Quality Stamp Content

Item	Contects	Type	Required	Example
Part-Information	appears once			
Part				
Name	Physical file name of CAD data	string	Y	SASIG-Example.prt
Revision	History of CAD data	string	Y	001
Size_KB	Size of CAD data（K-byte）	positiveInteger	Y	10752
CAD-System				
Name	Name of CAD System	string	Y	MyCAD
Release	Release(Version) of CAD system	string	Y	V1R02
Last-Save				
Date	Last saved date of CAD file in format yyyy-mm-dd	date	Y	2005-05-23
Time	Last saved time of CAD file in format hh:mm:ss	time	Y	14:10:53
Version				
Information1	*e.g. Part version	string	N	1
Information2	*e.g. Part GUID	string	N	84776d0f-5ebc-11d8-8747-00306e0a82e9
Information3	*e.g. edge-ID last-face-ID last-node-ID	string	N	12:6:Block3
Owner				
Organisation	The company name which made the CAD file	string	N	Company
Department	The department name which made the CAD file	string	N	Department
UserID	The username/ID which made the CAD file	string	N	ID
Check-Environment	appears once			
Checksum	Signature, fingerprint or other kind of manipulation protection information of the CAD file	string	N	2e67ed097c033e58848667c55ce95707
Check-CAD-System				
Name	(CAD) system name which performed the check	string	N	MYCAD
Release	Version of (CAD) system mentioned above	string	N	V1R02
Check-Tool				
Name	The tool name used for the check	string	Y	Mychecker
Release	The version of the Tool used for the check	string	Y	2.0
Check-Profile	The file name which showed contents of check execution	string	Y	SASIG-Profile
SASIG-Version	Version of the SASIG PDQ Guideline the checktool is conforming to	string	Y	2.1
Check-Date-Time				
Date	yyyy-mm-dd	date	Y	2005-05-25
Time	hh:mm:ss	time	Y	19:00:05
Operating-System				
Type	The operating system name when a check was done.	string	Y	MY OS
Version	The operating system version when a check was done.	string	Y	15.8
Check-Performed				
Organisation	The company name which excuted a check	string	Y	Company
Department	The department name which excuted a check	string	Y	Department
UserID	The username/ID which excuted a check	string	Y	ID
Responsible				

Organisation	The company name which is responsible for a result of a check	string	N	Company
Department	The department name which is responsible for a result of a check	string	N	Department
UserID	The username/ID which is responsible for a result of a check	string	N	ID
Check-Result	appears once			
Quality-Value	The synthetic evaluation point (It hasn't been written down since mention is difficult.) of a result of a check	nonNegative Integer	N	74
Number-Total-Entities		positiveInteger	Y	1500
Number-Checked-Entities		nonNegative Integer	Y	1400
Check-Log-File		string	N	SASIG-Example.result
Criteria	appears min. once and is repeatable			
Internal-Code		string	N	MCE-S09
SASIG-PDQ-Code		string	Y	G-SU-LG
Name	Criteria name in SASIG PDQ-Guideline	string	N	Large surface gap
Parameter1		string	N	0.02
Parameter1 Unit		string	N	mm
Parameter2		string	N	
Parameter2 Unit		string	N	
Parameter3		string	N	
Parameter3 Unit		string	N	
Number-Checked-Entities	The number of entities checked with this criteria	nonNegative Integer	Y	15
Number-Violating-Entities	The number of entities of violation of a check with this criteria	nonNegative Integer	Y	3
Remark	appears max. once	string	N	This is an example for the SASIG-PDQ Quality Stamp Version 2.1

9.5 How to Realize the Quality Stamp

Each check tool must be able to read and write a quality stamp interactively as well as in batch mode. Since CAD models might not always be modified by a check tool, two ways of quality stamp storage are defined by this SASIG-PDQ Guideline: outside of a CAD model as an external file, or inside as a kind of "attribute."

9.5.1 The Quality Stamp in the Form of an External File

The linkage between quality stamp and CAD model is made by the uniqueness of filenames, e.g., CAD-modelname = NICE- MODEL.model ,
filename of quality stamp = NICE-MODEL_model.xml.
With this definition it is possible to handle different filetypes. (e.g., .prt, .asm). Since the quality stamps are usually used during asynchronous data exchange between different partners, they should be transferred in a package together with the corresponding CAD files. In that case it is possible to ensure definite relation between the quality stamp and the CAD model. The storage of quality stamps in PDM systems is possible, but then it is the PDM system's task to store the link between model and stamp. Basically, you have to decide whether you want to store quality stamps in the PDM system, because check profiles are still under development today.

9.5.2 The Quality Stamp Inside the Checked CAD Model

The quality stamp is stored directly inside the CAD model readable for the check tool only. In some CAD systems, data may only be accessible and modified by a specific module; other systems might use other parameters or attributes. With this internal method, not only the strong physical connection of model and stamp is ensured but also the modification protection of the quality stamp.

9.6 Manipulation Protection

Although "manipulation protection" is important, the SASIG-PDQ working group does not recommend to use highly sophisticated authentication, e.g. via public encryption methods, yet. The check for validity of a quality stamp against its CAD model can be done at any time by re-checking the model. For that reason the timestamp for example may be used as a simple method.

CAD models usually have internally stored the date and time of last filing. The quality stamp shall, after a successful check, contain that last filing date and time. When modifying the CAD model (e.g. because of "Healing"), the check tool shall offer a combined functionality to modify the model and store the quality stamp. Hereby it is ensured that subsequent modification of the CAD model can be recognizedsed with the help of the quality stamp. For additional security, the file size is stored in the quality stamp, too. "Last file date and time" may not be changed during data exchange.

9.7 Checking a CAD model with a quality stamp

During a model check with a check tool, it should be optional selectable to read an existing quality stamp. The stamp will be checked regarding the consistency to the referred CAD model (check of timestamp and file size) and whether the criteria listed in the quality stamp are conforming to the required check profile. If those checks succeed, the check summary of the quality stamp will be displayed, if not, the model check shall be performed and the defacto errors shall be displayed.

9.8 Quality Stamp XML-file Example

An example for such a XML file can be found in Attachment I, its accompanied XML Schema file can be found in Attachment H.

10 Other Data

This section is intended to capture and present the data required and the quality of that data for other applications that may need to be considered in a product development environment. In today's product development processes, CAD data are being used in many aspects of the business that have never been able to directly use the data. This sharing of accurate, representative, quality product data allows for the greatest benefit to be derived from the data. This section will be completed in a future version of this set of guidelines. The following are possible subsections, providing a sense of what will be covered.

- Technical Manuals
- Illustrations
- Raster Data (**Note:** An ODETTE recommendation is available reg. raster data exchange)
- Material Specifications

SECTION III: IMPROVING PDQ

This Section provides concepts, methods, and guidance on how to go about improving product data quality. It addresses a wide variety of factors that contribute to product data quality problems that, if ignored, will likely prevent significant improvements from taking place.

11 Improving Product Data Quality

The total resolution to product data quality problems will take a long time, if ever, and will ultimately require the participation of all. This responsibility must begin with the initial design at the OEM or originating design environment and include all trading partners throughout the product development process, including the supply chain. There are, however, a number or steps that can be taken in the short term as well as the long term. The focus here is on the short-term, quick fixes that can make a difference right away. Any steps to improve the quality of the data can have great impact in the overall product development process and cycle. For instance, independent studies have shown that the Pareto rule may be able to be applied here. That is, addressing 20 percent of the up-front design flaws may eliminate up to 80 percent of the downstream re-work. Areas where improvements or impediments to improvements are likely to be found include the following:

- Readiness for Change
- Project Management
- Product Development Process
- Supplier Roles
- Cultural Drivers
- Skills and Motivation
- Communication
- Technology Base
- Reward Systems and Metrics
- Checking tools
- Healing

The following sections describe work that can be undertaken in these areas to improve product data quality.

11.1 Readiness for Change

Nine aspects of a work group or organisation are predictive of readiness to implement new technology. These nine areas, along with a brief explanation of their significance, are presented in Table 8.

For each area, a company can be categorised as:

- Favouring change
- Being neutral to change
- Favouring the status quo

Table 8. Dimensions of readiness for change.

Readiness Factor	Definition
Perceived need for change	The more people feel a pressing need to change, the more ready they are to accept it.
Track record with similar technology or process	The more familiar people are with the new technology or process, the easier it will be for them to adopt more of it or a different version of it.
Track record of change	Organisations that have a positive history of adopting changes are more willing to take the risks associated with bringing new changes on line; a history of failure, conversely, makes them gun-shy of the next effort.
Champion	A high-level supporter is almost always required for innovation to take hold.
Funding	Always helps. Resource-starved organisations usually skimp on support.
Organisational barriers	Barriers to informal communication within an organisation inhibit the spread of the informal knowledge required for successful implementation.
Communication among co-workers	A local version of "organisational barriers." Implementation of new systems requires considerable mutual help, for which cooperative work relationships are essential.
Job design	Having jobs designed to leverage the information-processing capabilities of new systems increases readiness for change.
Implementation process	Carefully thinking out the steps for bringing in new systems is important for their smooth adoption.

A company's readiness for change, or its resistance to change, derives from a number of structural factors, including size, market position, corporate strategy, and technology. Small companies tend to be more innovative than large, although at the same time a small company may have so little market leverage that it feels highly vulnerable and therefore risk-averse. Large companies have the market power to promote change but are often stymied by internal conflicts and requirements for internal communication (which really are equivalent).

Young companies are more ready to change than mature companies; mature companies have a strategy that focuses more on harvesting returns on investments made over decades rather than on making new investments. Related to this is a company's technology position: a company based on a mature technology, even if it is a young company, will have fewer change options than a company based on a rapidly evolving technology.

A profile of a change-ready company is a small company that has ample resources for new investment and whose current investments are in rapidly changing technology. A profile of a change-resistant company is a division of a large, mature conglomerate stamping out commodity parts based on old technologies.

Companies with many negative factors for change readiness will most likely resist larger changes. Smaller changes may be the best possible in those companies.

11.2 Project Management

Project management styles are important in supply chains. For example, a heavyweight manager in one firm, one who has immediate decision-making power, may become frustrated when dealing with a liaison manager, one without immediate decision-making power, in another firm. The heavyweight may interpret the liaison's lack of immediate action as "foot dragging" if the differences in project management are not well understood. A brief description of each of the styles, including strengths and weaknesses, is outlined in **Table .** **Table 10.** presents the typical responsibilities for each of these types of project managers.

Table 9. Definition of project management types

Type	Definition
Functional	Each function in a firm independently works on its part of a project. This works reasonably well in very small firms where informal communication across functions is high or on products that have functional subsystems that barely interact. This form of management is slow in its response to changes in the environment and requires a relatively slow, sequential approach to development with long feedback loops.
Liaison	This is the weakest form of cross-functional project management as it essentially retains a functional structure. This can be very useful for addressing short-term needs during a critical phase of the development process but often involves fixing problems that could have been anticipated earlier through a more powerful form of project management.
Lightweight	This often involves a design engineer or product marketing manager mainly responsible for such co-ordinating activities as sharing information across functional groups, setting project goals, scheduling, etc. The strengths of this form of project management are in focusing project resources on company goals while retaining synergy within functions. However, because of a lack of power in directly influencing people on the project, responsiveness and speed of development are not very high.
Heavyweight	Project managers directly supervise project members' work and may be responsible for their hiring and evaluation for the project, although overall performance evaluation and longer-term career development usually still rests with a functional manager. The benefits of this style include stronger employee loyalty and commitment to the project, strong co-ordination of different functional specialists, and a clear focus on the end goal. Drawbacks include the possibility of conflicts with functional managers who still must be managed from the top and the possibility that heavyweight project managers may take control of resources needed elsewhere in the organisation.
Autonomous	Project managers have full control over members of the project, including hiring, firing, and evaluation. Top management holds the project manager fully accountable for final project outcomes while allowing the project manager to create incentives and norms for employees in the group. This style has the benefits of putting complete control in the project manager's hands, in effect creating a small focused company that can act very quickly in a crisis. On the other hand, it reduces synergy between functions to near zero and, if it is applied on an isolated basis to one or a small number of projects it may result in a "rogue project" that loses track of both company and customer goals.

People who create and modify product data have the primary responsibility for product data quality. However, project managers need to support them in that goal by making available the time and resources needed to do a good job. For example, a CAD user under high pressure to produce a model, either because of tight deadlines or because of having to do more work than one person should be doing, will have difficulty spending any time ensuring that the quality of the model is high. To do the job right, there must be enough CAD users available, they must have adequate skills, they must have the right software, and they must have enough time.

In most situations, the project manager is responsible for making sure the conditions for product data quality are met, from the resources to the communication of product data requirements, if not the actual negotiation with the trading partner. The more authority and resources a project manager has, generally the easier it will be to provide that support. For example, if the requirements of a project force significant variations from a company's established procedures or standards, a heavyweight project manager is more likely to be able to protect CAD users from pressure to follow the "usual" process.

Table 10. Project management responsibilities

	Functional	Liaison	Light-weight	Heavy-weight	Autono-mous
Share technical information among project members in different functions		X	X	X	X
Distribute reports, minutes of meetings		X	X	X	X
Set project goals			X	X	X
Schedule and co-ordinate project activity			X	X	X
Allocate funds and equipment for project			X	X	X
Select staff for project				X	X
Evaluate project performance of members				X	X
Evaluate overall performance of members					X

11.3 Product Development Process

The nature of the product development process varies widely from company to company as well as from supply chain to supply chain. Some companies have very detailed and formalised company-wide design processes. Other companies set a few high-level milestones into a schedule and then allow the various groups involved to determine their best process to meet that schedule. Regardless of where companies fall in that range, meeting the schedule is generally very important.

Individuals actually doing product development do not always follow their own company's policies and procedures, resulting in much greater variation than might appear on the surface. However, much of this individual variation is in fact a result of attempts to meet difficult schedules or to otherwise adapt to working with their customer or supplier—to make their two processes "fit" in order to get their jobs done effectively.

The overwhelming importance of the schedule results in engineering taking a view of costs that sees additional costs, expressed as engineering overtime, as sometimes necessary to meet schedules.

Schedules, budgets, and requirements are established at corporate levels, and then rolled down to the various engineering components. Of the three, schedule is the least negotiable, so much so that overtime is routinely applied to meet schedule milestones. If this is ineffective, features and performance targets may be modified to meet the launch date.

The specific processes used within companies are less interesting than how the design processes of the different companies interact to form a process for the development of the overall product. As might be expected, the customer's delivery requirements within its product development process tend to drive the supplier's process.

Coupled with the supplier's role, the product development process determines when exchanges should take place. This includes when, in terms of the stage of product development, as well as how frequently exchanges are necessary.

> *The company has a design process* – This assumes the company not only has a formal design *process* but that it is well defined and people actually understand and use it. Within this process, determine when data will need to move from one trading partner to the other. Reaching agreement on this before the work starts will let everyone know what to expect and what is expected. Such a mapping need not be exact to the point of specific days that data will move. Rather, it should lay out the requirements in a form such as the following:
>
> > "During the conceptual design phase, which will run from the beginning of the project for the first two months, product models will be sent from the supplier to the customer on a daily basis. During the detailed design phase, running from the end of the conceptual design phase for six months, the supplier will send updated product models at least once a week plus whenever any change in the overall envelope occurs. In addition, current product models will be sent immediately before each of the scheduled reviews."

The process should also specify who is responsible for what during the design process, including individual accountability for data and data quality.

> *The company lacks a design process.* This includes (1) the situation where a formal process is *absent,* and (2) the situation where a formal process exists but most people ignore it. In either case, determining a schedule for data exchanges in advance is more difficult. It may still be possible to establish a schedule for exchanges as described in the previous example, but the timing for shifts in frequency and reviews will likely be much less predictable. Keeping both the customer and supplier happy will depend greatly on close co-ordination between the two companies.

The best approach in either case is to establish an ongoing team of people from the two companies. This team will oversee product data exchange timing and designate responsibility in both companies. This should be a natural follow-on to the discussions described under "Supplier Roles" in Section 11.4.

Another issue within product development is whether the life cycle of the product data is understood. The people doing product design need to understand how the data they create will be used and the data requirements for those uses. Specific data requirements depend on how the data will be used. If data are unusable for a given purpose, then their overall value is reduced.

11.4 Supplier Roles

Table 1 shows an approach to categorising a supplier's role. The table shows increasing levels of involvement as the supplier moves from left to right. The seven characteristics used to define the supplier role are the following:

- *Design responsibility* - Who is responsible for designing the part or component?
- *Product complexity* - How complex is the "product" being supplied?
- *Form of specifications provided* - How are the specifications provided to the supplier?
- *Supplier influence on specifications* - How much influence does the supplier have on the specifications?
- *Timing of supplier involvement* - When in the product development process does the supplier get involved?
- *Component testing responsibility* - How much responsibility does the supplier assume for testing the part or component?
- *Supplier product development capability* - What resources does the supplier have to develop new products in house?

Table 11. Supplier roles in product development[1]

	Contractual	Consultative	Mature	Partnership
Design Responsibility	Customer	Joint	Supplier	Supplier
Product Complexity	Simple Parts	Simple Assemblies	Complex Assemblies	Complete Subsystem
Form of Specifications Provided	Complete Design	Detailed Specifications	Critical Specifications	Concept
Supplier Influence on Specs	None	Present Capabilities	Negotiate	Collaborate
Timing of Supplier Involvement	Prototyping	Post-concept	Concept	Pre-concept
Component Testing Responsibility	Minor	Moderate	Major	Complete
Supplier Development Capabilities	None	Little	Moderate	High

[1] This table is derived from an article entitled "A Second Look at Japanese Product Development," by Rajan Kamath and Jeffrey Liker in *Harvard Business Review*, 72, 154-170 (1994).

The supplier's role in product development plays a major part in determining the data exchange requirements and sets the stage for most of the remaining solution areas. For any given supplier:

1. *Determine the role for that supplier.* Decide what type of role that supplier currently plays in product development.
2. *Determine basic data exchange requirements.* A team of representatives from both the supplier and the customer should determine what kind of data will need to be exchanged by the supplier and customer, including the direction of the exchange (if the requirements depend on who is sending to whom). These requirements should be based on the supplier's role and the complexity of the part or assembly the supplier will be making. In addition to managers, engineers, and sales/purchasing staff, the team determining the requirements should include product data users from each company because they know best what the real data requirements are and how to meet them. Items to address include the following:

   - What kinds of product data exhibit quality problems?
   - What form of product data best supports the given supplier role (e.g., solid, surface and wire frame, fully annotated model)?
   - What level of detail should the product data include (e.g., fully detailed, limited detail, envelope, assembly)?
   - Which company is responsible for adding characteristics to the product data that are required for manufacturing (e.g., material selection, draft angle, shrinkage allowance, and spring back)?
3. *Establish detailed standards for the form of the data to be exchanged.* By establishing such standards and sharing them with trading partners, companies can provide software users with very important guidance on building high quality models. Ideally, most of those requirements will already be captured in existing formal CAD model and other product data standards. However, that may not be the case. There are three possible situations:
   - Both companies have formal product data- related design standards. If both companies have established formal product data related design standards, such as CAD data standards, there needs to be an accommodation process. Lacking an industry standard as a model, such internal standards are likely to conflict. Too many of the decisions are arbitrary. If a company has established rigid standards based on only its own needs, then the trading partner may not be able to effectively use the product data that result. An accommodation between the two standards needs to be negotiated.
   - Only one of the companies has formal product data standards. If only one of the trading partners has established formal product data standards, then its standards will probably become the basis for exchange decisions. The negotiation process will likely be straightforward.
   - Neither company has formal product data standards. If neither company has established product data standards, then negotiation for how product data will be built for the specific project should take place. In this case, the discussions should focus on the common data aspects. For example, if only solid models will be exchanged between CAD systems, then standards that isolate the solid model and apply to it are all that need to be agreed upon. However, if models with details of manufacturing or other kinds of information will be exchanged, then conventions for layering, colour, dimensioning, etc. need to be addressed and agreed upon.

11.5 Cultural Drivers

Company culture can be an aid or an obstacle to improving the quality of the product data produced in the company. Assessing the separate corporate cultures of trading partners is important both to understand how well the relationship between companies will prosper in a new technological regime and to assess the capability and readiness of each partner to implement the new technology. Corporate and functional cultures can either facilitate or impede communication and co-ordination between trading partners. Organisations that are highly similar in culture will generally have an easier time co-ordinating their activities; they "understand" each other. Organisations that are highly dissimilar in culture will have a harder time forming any but the most contractual, arms-length relationships. Though they speak the same language, they may still have trouble communicating.

In the absence of a formal cultural appraisal you can still make a useful, if superficial, assessment about the impact the organisation’s culture will have on improving product data quality. Organise a cross-functional team of people who create, use, or transfer product data and ask them the following questions:

1. Do people in the company perceive status differences between people who create product data and those who use product data?
2. Do product data creators tend to ignore the needs of data recipients or see the recipients as adversaries?
3. Do people in the company believe that the quantity of what they produce or the speed at which they produce it is more important than the quality of the result?
4. In general, do people in the company see limited value in teamwork?
5. Are schedules and deadlines regularly ignored?

For each question, decide where the preponderance of evidence lies. If the tendency is to answer “yes” to several of these questions, a cultural problem may exist that will work against improving product data quality. These problems are not easily or quickly overcome. The best simple approach is to train people to change behaviour that manifests itself from these attitudes. For example, if people tend to look down on the recipients of data, that may manifest itself by not delivering the data when promised or delivering data that has known problems. Make it a policy that product data must be delivered on time and without errors, and train people to treat the product data recipient as the internal “customer” he or she is.

If reason is applied and communicated as to why the quality of data needs to be improved, people generally want to do a good job. If expectations, responsibilities, ramifications, and accountability are all clearly defined and enforced, the design community will understand the impact of their design and the quality of their work. One of the last things a designer or a product development manager wants is for his/her design to be changed by a downstream user. With the appropriate product data quality, the need for modifying the data downstream is greatly diminished and possibly even eliminated.

11.6 Skills and Motivation

Both the skills and motivation of the people involved are critical to product data quality. For skills, the creators of product data need to know the characteristics of high-quality product data. They must also know how to design and build product data, such as CAD models, so that those characteristics appear in the models that are produced. Training is usually required in the following situations:

- If the company has recently adopted or modified formal product data standards.
- If product data content requirements have been negotiated with a trading partner.
- If CAD or other software users are not fully conversant in the capabilities of their systems.

- If new supporting tools or techniques has been implemented that involve the creation or use of product data.

Motivation to produce quality product data is also a critical issue. People need to understand that they have goals related to product data quality. If, for example, they understand how critical product data quality is to the success of the product, and if one of their goals is the success of the product, they will be motivated to improve their work. While this may seem obvious, it often is not obvious to lower level staff. They need to clearly understand why they should be motivated to change their usual way of doing things to result in higher-quality product data.

In general, people want to do a good job, but they must know what doing a "good job" means and have the necessary tools and motivation to achieve that goal. These motivations can come in many forms but must include management buy-in from the top as well as ways to enforce policies such as criteria checks at specific release gateways, or requirements for specific disciplines or suppliers.

11.7 Communication

Exchanging product data is a form of communication in and of itself. The purpose is to provide a useful description, in the form of a mathematical model, of a part, set of parts, or assembly. Another way to consider product data quality is to think of it as a responsibility of the sender in the same way any other communication would be. In written communication, whether a book, an article, a letter, or an email message, it is the writer—the “sender”—who is responsible that it be properly understood. Product data are much the same. It is the responsibility of the person who creates the product data—the “writer”—to ensure that the data will be understood. Such an understanding requires that the product data be complete and correct. This concept also presumes that the exchange process itself does not cause problems.

As in other forms of communication, the recipient—the “audience”—is an important part of the interaction. If you are sending data, what does the other person really need? If you are receiving data, have you communicated to the sender what is needed, not necessarily what is wanted? People often want to receive more information than they need and are also often disposed to send less than the other person needs. Somewhere between is the reasonable balance point. To determine that point requires that the data sender and the data receiver sit down in advance and negotiate what the content will be. They need to have clear decisions and agreement on expectations, responsibilities, and reasoning for each aspect of the product development process. Technology, processes, and policies can be incorporated into the product development process and environment to ensure that each and every participant understands his/her role and the reasoning behind the data requirements and the process flow. This discussion should also be included in the initial negotiations described in Supplier Roles (Section 11.4).

11.8 Technology Base

While product data quality depends heavily on CAD and other software users, there are some technological solutions that can help. Many CAD data quality problems are basic problems with the mathematical representations that make up the model. For example, in wire frame models, lines that are supposed to meet do not. In surface models, surfaces might not be properly trimmed or bounded. Solid models might end up with faces with structural problems. Any kind of model might have excess data, such as duplicate versions of the same part of a model. Any of these can cause problems for CAD data users.

Most major CAD systems have various checking tools built in that can be used to verify certain characteristics of a CAD model, such as whether it is a valid solid model. If available, those tools should be used frequently during the development of a CAD model to catch problems as early as possible and again before sending a model to someone else.

Additional CAD data quality software checking tools are becoming available from third party vendors. These tools allow further checks before a model is shipped. Some work on the original model, whereas some work on models that have been translated into neutral formats such as STEP or IGES. Since poor quality CAD data often leads to delays and additional labour costs, such tools have the potential to pay for themselves rapidly. Of course, using any such CAD data quality tools depends on the knowledge of the user. Therefore, there will always be a training aspect to implementing data quality.

11.9 Reward Systems and Metrics

Product data quality will not improve unless product data system users, the engineers they work with, and their managers are rewarded for producing high quality product data or deterred from producing poor quality product data. One of the common problems in design groups is that the reward systems are geared to reward fast production of product models. If the rewards are focused on speed, speed will result, but data quality will suffer. This conclusion is based on talking to CAD users who have explained how they were caught in exactly that situation. Although they knew how to produce higher quality models, they were effectively punished for doing so because it took more time. Hence everything they did was oriented toward fast production of what amounted to “pretty pictures” rather than effective CAD models. This, of course, implies that top management be aware of the direction to implement CAD qualityand understand the ramifications on deadlines as well as the benefit from shortening the overall product development process and cycle.

The reward system must be balanced so that product data quality is a significant part of the equation. Dividing rewards appropriately between individuals and groups must provide part of that balance. Group rewards can be especially useful when addressing issues that affect relationships between groups. Since product data often cross group boundaries, product data quality is a good example of a situation where group rewards are likely to be an important part of the solution.

An example of deterrence is to put a check in place prior to specific design gateways, e.g., release. This would set up a process where each model to be released must be checked, flagged, fixed, or overridden to go into the release phase of the process and the release database. If such a policy were enforced, it would require the necessary information and quality in the product data for that particular gateway, discipline, or supplier's needs.

Of course, a critical aspect is how product data quality is measured. Unfortunately, there is no perfect method. What makes high quality data depends in large part on the uses to which that model will be put. One can say that an accurate model is always going to be necessary, that is, the data that are provided need to accurately reflect the end product. For CAD data, that means that all the geometric entities that make up a CAD model are in the correct positions to the intrinsic accuracy of the CAD system. For example, a classic problem that has long been seen on manually created drawings is where someone has changed a numeric dimension value without actually correcting the part drawing. The result is that the drawing graphic no longer matches the intended real item. This same shortcut can be used in a CAD system, with the potential for even more drastic errors if the incorrect model is used as a starting point for other activities.

As with many of the other solutions addressed under product data quality, there is a clear need to negotiate the metrics that will be used to determine product data quality as part of defining product data exchange requirements. While the following metrics will not always be appropriate, they serve as examples.

For all models:

- Layers and groupings exist and are the same as the original.
- Parametric information is present and correct.

For solid models:

- Minimum area of surfaces defining faces
- Unusually large variation in surface areas (ratio of largest to smallest surfaces)
- Maximum gap between surfaces
- Presence of very narrow (unusually high length-to-width ratio) faces with four boundaries
- Whether sections can be cut

For surface and wire frame models:

- Whether possible to create a shell
- Duplicate geometry
- Twisted surfaces (flipped surface normals)

In part because an absolute list is difficult to provide, trading partners should negotiate what metrics they agree are important and then enforce those metrics contractually.

11.10 Checking Tools

One of the ways to approach CAD data quality is to introduce technology that identifies the flaws in the CAD model. The automotive industry has seen a surge of interest in this technology. Software products specifically address model quality and interoperability issues. There are also functions in most CAD systems that help identify the problem areas of a model.

Implementing a model quality-testing program can yield substantial product data quality improvement. Validating CAD files prior to release significantly reduces model rework time. Industry estimates have shown that model rework time can be cut by 50 percent in downstream product data exchange and numerical control (NC) applications. That number can be as high as 80 percent for rapid prototyping and finite element analysis functions. As CAD models continue to take on a broader and more significant role in the development of new products, it is important that these files flow smoothly into downstream applications.

Commercial software now exists that supports CAD model quality. Some applications analyse CAD models, detecting problems that may lead to problems in downstream applications. Once identified, these geometry or topology problems can be resolved by the designer in the early stages of development where changes can be incorporated quickly and with little cost. The downstream user is not always able to send the model back to the designer. Therefore, these tools can also be employed by the downstream user to identify the areas of the model that may be causing a problem.

An extensive, ongoing, iterative study will need to take place to determine which flaws and of what size cause problems. Which downstream applications do these flaws cause a problem for? Because of different products, applications, processes, and methodologies, this study will need to take place in each unique product development environment that is being considered for implementing CAD quality. This study will continue as new products, applications, technologies, software revisions, etc. are introduced to the environment. This study will need to be iterative to determine precise values and conditions that cause problems for the downstream users. This process is the most crucial in implementing a CAD quality initiative. It is imperative that these flaws are proven to cause problems in downstream applications and that the designer and the CAD application have the capability to address the problem. In other words, there is no need to identify a flaw that doesn't cause a problem or that a designer cannot fix because of the CAD application's inability to do so or the designer's lack of knowledge.

Once these values are identified, tested, confirmed, and documented, ideally they can be set up in the CAD quality-checking tool, identifying and presenting these flaws to the designer. This allows the designer to anticipate system restrictions and ensure that models created will flow seamlessly into all downstream applications. In short, this allows unrestricted interoperability to be designed into the model.

These tools are being widely accepted in the industry as a big part of "today's answer" to implementing a CAD quality initiative. However, it is important to note that to do this correctly, the CAD quality software is only a small portion of the equation. Often the first reaction is to use these tools as a checking tool, only checking the CAD model at the point of release, or checking the quality of the CAD data coming from suppliers. Although this checking concept should be incorporated as well, the real value is to check the model during creation, not wait until the model has been completely finished and detailed. This is like turning on a spell checker after typing a 100-page manuscript. A robust implementation of CAD quality must be implemented throughout the product development process. This includes primarily up-front design but also as a checking tool at certain gateways along the product development process, as well as a tool for the downstream user to identify and fix the flaws as needed. To realise real impact on the data flow, the product development process must include the following considerations (some of which are elaborated elsewhere in this section):

- CAD Application Capabilities
- Constraints - Does the CAD application being used allow for fixing the flaws being identified?
- Enhancements - Can the CAD application be enhanced?
  - » Bugs - Is there a bug in the software that continues to cause the flaw to show up?
  - » Import/export functionality/robustness - Is there a robust way that the CAx applications can share data? Are there issues with the exchange or transfer process itself?
- Reactive Solutions - Fixing the problem, once identified
- Preventive Solutions - Enhancing the CAD application or designer skills so that the flaw does not happen in the future
- Designer Skill - Do the designers have enough training? Are they familiar with techniques to fix the flaws? With methodologies to prevent the flaws?
- Training/Mentoring - Are the techniques and methodologies being incorporated into the user training? Are there resources available for existing designers to get update training? To get instant answers to solve a problem? To prevent a problem?
- Methodologies/Techniques - Are techniques and methodologies being captured to fix and prevent problems? Are these being communicated appropriately? Are they being incorporated into CAD support? Into training? Into help resources?
- Resources for Help - Is the CAD support group able to support the designers with these techniques/methodologies? Is there help available for the designer? On-line? Hardcopy? Phone support?
  - » Feedback Communications - Establish a feedback loop from the downstream users to CAD support and the upstream users.
- PDM Considerations
- Checking prior to submission
- Flagging to acknowledge that the model has been checked

- Pass/fail flags, fail with explanation, and override?
- Product development progression gateways - concept, pre-release, work-in-progress, release, etc.
  - » Associating CAD quality results files with appropriate versions of the CAD model
- Consider model content at each gateway--What is acceptable? Required? Causing problems? Unnecessary?
  - » Tracking/Monitoring CAD Quality
- Diagnostics/Values - Update the need for certain checks and the values of those checks as new technology, methodology, products/features, and requirements are incorporated into the product development environment.
- Impact on product development process - Closely monitor the impact of these technologies, processes, and policies on the product development process.
- Too-stringent requirements may bottleneck data flow.
  - » Model release deadlines may be adversely impacted by having the designers clean up models. This may be acceptable if the downstream users can realise a great benefit by the clean models; therefore, this must be tracked also. This may prove to be more difficult as the downstream users may be diversely located or actually in different organisations and/or companies.
- Cost vs. Savings - Although sometimes intangible, and almost always difficult to capture and quantify, it is necessary to be able to monitor the cost vs. savings of implementing a CAD quality initiative. Ideally, this will show the very positive impact of quality data, allowing for the implementation of technologies, methodologies, processes, and policies to enhance CAD quality.
  - » Track what the cost of implementing CAD quality is:
    - Training
    - Designer time spent fixing
    - Checking procedures
    - Models being denied for submission, returned for repair
  - » Track what the savings of implementing CAD quality are:
    - Increased overall productivity
    - Downstream ramifications
    - Incorporating downstream feedback into design more robustly (i.e., CAE, prototyping, etc.)
    - By implementing techniques, methodologies, and software enhancements, are the designers being more productive?
    - Do the designers better understand the downstream user's requirements?

Model quality testing provides a practical solution for CAD/CAM/CAE software users throughout the industry and bridges the interoperability gap. This enables CAD models to continue to take on an expanded role throughout the product development process.

11.11 Healing

11.11.1 Introduction

Healing means to rectify errors in product data. Current healing is generally carried out in the final stages of the design process, and there may be a substantial time period between when the design has been completed and when the need for corrections is noticed. During this period, not only parts of the design objectives fade away. The part may also have left the system in which it was generated, e.g., by STEP or IGES transfer, reducing design supporting information if existing.

Before any errors may be corrected, they must be found. This necessitates the use of a checker in order to detect the errors. While in principle a checker and a healer may be separated, if they are, there must be some mechanism bringing the errors detected by the checker to the attention of the healer. With two different pieces of software communicating with each other, there needs to exist a standard for how this will be accomplished.

To the best of our knowledge, a standard for transferring information from a checker to a healer does not exist today. Instead most often, if not always, these pieces of software are brought together in a unit. Clearly no error can be corrected unless it is detected, so the quality of the checker is a first limitation of what may be healed.

Another limitation comes from the fact that current healers address only a (small) subset of geometric quality criteria encountered in these guidelines. While a few of the errors that the healer handles may be satisfactorily cured automatically, often considerable manual work is needed. In some cases, this task is difficult, if not impossible, because tools normally available in CAD systems are missing. Even if the right tools are available, it can still be a demanding task for the normal user to identify, understand, and repair potential failures in the CAD model. Using an existing healer also means that the user must learn and maintain a working knowledge of one more system.

In the first section, we will discuss some of the shortcomings of current healers in more detail and with examples. We will also consider the mean effect of deferred detection and correction of failures, akin to the current healing workflow, allowing chains of errors to develop. Taken together, we see a severe inherent limitation of current healing processes.

To avoid the difficulties hindering an efficient and satisfactorily correction of errors, a new healing concept is needed. As a first step in this direction, we sketch a concept in the next section. Its cornerstones are detection and correction of failures as early as possible to fulfill the geometric quality criteria treated in these guidelines. It means that checks and removal of detected errors are firmly integrated into the design process, keeping the model correct during the entire process.Another important part of the new healing concept is to initiate a subdivision of the geometric quality criteria into groups depending on the degree of user interaction needed to correct the corresponding failures. This is initiated in the third section. But let us start even earlier, looking at CAD software as a source of some of the shortcomings, presenting suggestions on how they may be avoided.

In the first category are those failures that may be avoided by a suitable design of the CAD software. In the next category are shortcomings that may be reliably corrected without the user. Next are those requiring small to moderate user activity.Finally we have those in which the user must at least govern the entire correction and, perhaps, even perform manual correction. In general, we will exemplify with curves and surfaces rather than solid models,for example, both for simplicity and because several difficulties with solid models are generated at the curve- or surface level.

11.11.2 Current healing and its shortcomings

Software aiming at the restoration of defective CAD models is generally termed *healers*. From their marketing, one often gets the impression that using them is simply a “press the button activity” on the part of the user and without further involvement by the user. The geometry is automatically corrected.

To correct these misconceptions, let’s start with a real-lifecase. Figure 4 shows a fuel distribution pipe consisting of 34 636 IGES entities in which there are about 8 000 curves and 2 000 surfaces. The file comprises 24.2 MB and is created in a CADsystem that we call System A.

Upon importing it into a different CADsystem that we call system B, a log file consisting of 51 A4 pages is generated, of which 50 pages is an account of errors of different kinds. As an example, 100 errors of type 142 (Curves on a Parametric Surface) was encountered with a consequence that these objects were not processed.

Investigating the geometry that was processed with the internal quality checker in System B, one finds two main types of error:

Type 1: Original import surfaces have been split into pieces due to G2 discontinuities.
Type 2: The import geometry contains small edge(s) and imprecise vertices.

Small crosses in Figure 4 display some of the locations of type 2 errors.

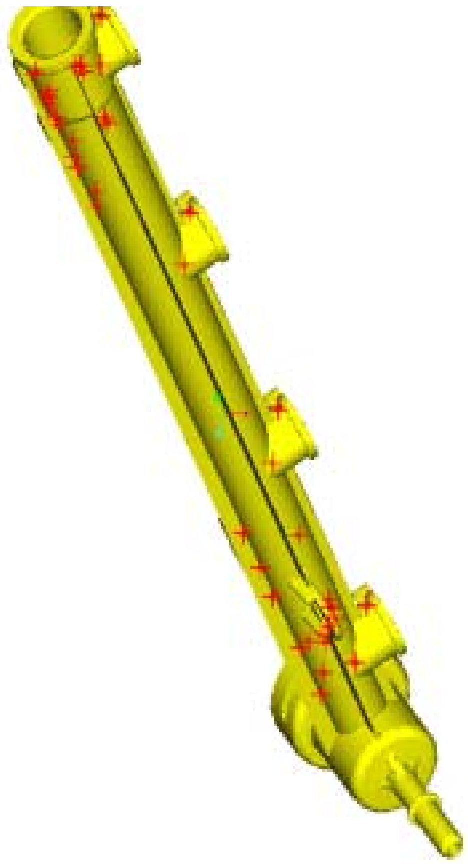


Figure 4. Examples: Location of type2 errors

What are the common ways today to take care of objects like this? In addition to possibly using healers, which seems to be rarely disclosed, we are aware of four different ways.

- Downstream users of the model to enable them to come any further with it do some kind of emergency fixes. For instance, it may be fixes enabling software for the injection moulding simulation to be run on the model. These users lack the design history and cannot correctly decide which parts may be changed and which should remain unchanged. They are not really repairing the model but removing the worst obstacles hindering their use. The changes they are imposing are rarely carried back to the original.
- Local CAD–design consultants, who are designers themselves, are fixing some or perhaps all of the errors of the model. In some cases, the consultant may perhaps infer the design intent or trace it back, but it is an error-prone and expensive process, and it cannot be taken for granted.
- An increasing number of global healing consultants are offering their services, often over the Internet.
- Remodelling the entire CADmodel by a local designer.

Even if the work to carry out the changes is moved in the three latter cases, the responsibility to determine whether the changed model is suitable for its intended purpose stays at the customer and it may well be a formidable task. Facing the situation in which a CAD model cannot be used for its intended purpose and, in analysing it, the CAD quality check software detects and reports some thousands of errors, it is very easy to become resigned. In this case it can be tempting to rely on an "automatic" solution. Common to the marketing of most healing systems is forwarding the extreme ease of use of the products. One may easily get the impression that healing is done more or less automatically. This may be the case in some very special situations:for example, having already spent some time on setting flags suitably for healing one model,if the task is to heal a very similar detail, the previous setting of the flags still makes sense.

Further, one must be prepared to largely accept the default settings of the vendor without questioning whether it is correct, or even reasonable, or not for the current task. The result of this kind of repair is of course highly disputable. Indeed, it is not uncommon that while improving the model with respect to some of the geometric quality criteria of these guidelines, huge numbers of violation against others are brought in.

If one looks for some slightly more responsible way to heal, much of the suggested ease of use fades away. It turns out that even with many (too many) default settings, one must often carry out the operations in different passes and a proper order.

Another common feature of healers is the almost total lack of description of how the repair is being done. Apparently the healers are not designed to primarily allow the user to firmly govern how the healing is carried out or to assess the result obtained. User interfaces allowing it are simply not there. In the cases where several possible solutions exist, as is very often the case, one possibility is picked out due to some general principle that is never disclosed.

Since healers are changing the model, it is evident that in all serious repairs, the user must know precisely what is going to happen when running a healer. The vendors should pay attention to solving the problem of how to describe exactly what their healer does in a user-friendly way. At present, their descriptions are so vague that there is no chance for a user to figure out which of several different possible solutions are used.

One may easily anticipate one vendor objection against these requirements: considering a defective model with many errors, like the one presented in the beginning, it would not be easy going for any user to get a sufficiently deep knowledge of each of the errors so that he/she could govern in which way to repair them. Even if the user happens to be the designer of the model, after some months, it is not easy to recall the intent of the design and infer what parts to change and what not to change.

Concerning the possibilities for the user to govern what will happen during the correction of the model, we must also consider the number of errors that appear. Reasonably, a user will only be ready to exert firm control on how to correct some ten or twenty errors. It calls for a new workflow in the use of healing, leading us to the next section.

11.11.3 A new healing concept

As we have already seen, current ways to heal exhibit such severe problems that this workflow ceases to provide a feasible solution of the task reliably repairing defective CAD models. Since creating correct CAD models is a very important task, with significant economical impact, we will look for an alternative workflow that addresses this possibility.

In the proposed workflow, the user--frequently and incrementally during the modelling--checks his/her CAD model with a native CAD quality checker. As soon as an error is detected, it is also being corrected. Here, an enhanced kind of healer would be very beneficial. When appropriate, it exhibits a sample of different ways to heal and automatically carries out the kind of repair the user has chosen. Finally, it displays what has been done, enabling the user to accept or reject.

In most cases, to generate flawless models, one needs to exercise firm control of the way the correction is done. To be able to do it, one must act before long chains of errors have been built up. It is not manageable to do careful corrections for more than some ten or twenty errors. It is hardly economical either, since generally many errors are secondary effects caused by primary error, i.e., they shouldn't have appeared at all if the primary errors had been cured in time.

From our study of the healers and based on the knowledge about what kind of CAD solutions that are currently feasible, we are convinced that this scenario is possible to realise in the near future. Of course, the code still missing must be written. A more challenging problem, however, is to present an error to the user in an efficient way, enabling him/her to form the right decision on what to do.

To realise the above procedure, it is vital to tell the CAD vendors what our requirements are and to indicate ways to reach these goals, encouraging them to use their expertise in providing the solutions.

Let us consider a couple of simple examples of these requirements. The first concerns a large segment gap (3.1.1.1. Large segment gap (G_0 discontinuity): G-CU-LG).

One potential solution would be to move what appears to be a quarter-circle in the direction of the upper line until both of its endpoints reach the two lines and then remove the overlap with the lower line by trimming off that line. It would be very hard for a healer to automatically detect that this may be a good solution. It may also happen, however, that the position of the quarter-circle is correct and that only the lines may be changed, which a healer cannot possibly decide itself, and this case calls for quite a different treatment.

While there is an easy solution for the upper gap in this case, the left one exhibits many potential solutions. Which ones of these that may actually be used depends on what may be changed and what may not, and in which way, information impossible for a healer to supply automatically.

One common current healer-solution is to bring both the encircled end points to their mean value, which may work for some purposes but destroy both curves for others, for instance, if the curves are styling-dependent. It may well be the case that one curve is correct and that only the other should be changed to join them with the desired smoothness. It should not be difficult to make the right correction while building the model and the requirements of the curves are well known--much more difficult after a while, when this information is much harder to recapitulate.

The second example is about non-tangent patches (3.1.2.2. Non-tangent patches (G_1 discontinuity): G-SU-NT).

Clearly there are many different ways to cure the problem and with very different outcomes. Again, which to choose depends on what one wants to obtain and what one is allowed to change. With this information available, it would be rather easy to direct the healer to form the desired joint.

A common way to “solve” the problem in the current healing practice is to replace the sharp joint with a narrow patch that connects tangentially to both the adjacent patches. There are some problems with this solution. First, it is a big risk that may generate a Narrow surface or patch: G-SU-NA as described in 3.1.2.10. This is often seen in practice. It may also create Surface with a small radius of curvature : G-SU-CR as described in 3.1.2.14. In addition, this may lead to the effect that while the connection is tangential in a mathematical sense, the connection is still experienced as non-tangential.

11.11.4 Grouping of the criteria according to needed user interaction

In order to heal violations of the criteria in the guidelines in the right way, one needs to know for each criterion which information is indispensable to make the correction. In building a healer, one must design it in such a way that it relies on this information, not on some general assumptions on what this kind of information may be. Such non-specific information may give reasonable results in some cases but a totally wrong outcome in others. An unpredictable result is not compatible with careful healing and must be avoided.

For this purpose, we propose to split the geometric criteria into the following groups:

(a) Errors that may be prevented from occurring by proper software design. We present this group, with examples, in more detail below. This group should be cured through proper software design and not burden the user.

(b) Errors for which no or very little user interaction is needed. Examples: Inconsistent edge on curve: G-ED-IT, Inconsistent edge in loop: G-LO-IT, Inconsistent face on surface: G-FA-IT, Embedded surfaces: G-SU-EM, Unused patches: G-SU-UN. For this group, currenthealing concepts may work well or may be enhanced to do so.

(c) Errors for which some but not heavy user interaction is needed. Examples: tiny curve or segment, inappropriate degree linear curve, relatively narrow neighbouring patches, etc.

(d) Errors for which a user must decide which method among several to use for curing the error. Examples: Filling gaps, increase smoothness etc.

Group (c) requires from extension to heavy extensions of present healer ideas, whereas group (d) requires new ideas.

Since healers are changing the model, it is evident that in all advanced repairs, the user must know precisely what is going to happen when running a healer. The vendors should pay attention to solving the problem of how to describe exactly what their healer does in a user-friendly way. At present, their descriptions are so vague that there is no chance for a user to figure out which of several different possible solutions are used.

In situations where several different solutions exist, it is not enough, however, just to tell the user which one of them is used as default. Instead, one should exhibit all the possibilities the healer provides. This sample should be sufficiently rich to cover the most frequent cases, and the user should be allowed to select which one to use.

We conclude by giving examples of geometric quality criteria that belong to groups (a) – (d) above, with some suggestion on how they may be cured. We start with those that would be possible to fulfill automatically by suitably designed CAD software. We envisage that the values decided upon are supplied in a set-up file for the CAD system, e.g., on a company base, with the effect of preventing objects not in line with the company's demands from being created.

Comments on (a)

3.1.1.4. High-degree curve: G-CU-HD. The maximum degree of any curve created by the CAD system would be possible to set. It would guarantee than no curves with a degree exceeding the maximum degree will be created.

3.1.1.5. Indistinct curve knots: G-CU-IK. The minimum non-zero difference between consecutive knots would be possible to set. It would guarantee that no curves having a minimum non-zero difference below this limit would be created.

3.1.1.6. Self-intersecting curve : G-CU-IS. A self-intersection may occur unnecessarily, simply because the software does not protect against it. By using constrained optimisation, this failure may be avoided. Since the latter requires longer execution time, we recommend it be reserved for cases in which the normal procedure fails.

In the case of creating offsets to plane curves, the creating software would carry out the analysis of whether the offset distance exceeds the inside radius automatically and inform the user that a useful curve could not be defined with this distance. It should also provide the user the maximum offset distance for avoiding self-intersection, and in the case where the smallest curve radius of curvature is set, (see 3.1.1.9. below), the maximum offset distance compatible with it.

This is the information that can be obtained automatically. Knowledge of it enables the user to decide how to proceed, but the decision of how to handle the situation must be up to the user.

If, for example, a desired offset distance to a curve is larger than the maximum offset distance from the analysis, the originating curve must be modified.

3.1.1.9. Curve with a small radius of curvature: G-CU-CR. The minimum radius of curvature of any curve created by the system would be possible to set. By using constrained optimisation, the requirement may be met. Because of higher execution time, this should be reserved for curves violating the requirement.

Comments on (b)

Basically, these criteria concern switching between two well-defined cases or deletingobjects and are handled rather well by current healers. The desired improvements consistof increasing the user control over the process to prevent objects from being unintentionally deleted, always including the possibility to undo the operation.

Comments on (c)

3.1.1.10. Tiny curve or segment: G-CU-TI. A healer can automatically provide the three natural extrapolations. The user selects which of these to accept or elaborate further, with healer tools or tools from the native CAD system.

3.1.1.12. Inappropriate degree linear curve: G-CU-ID. A healer can very easily generate the approximating line and check if it is within tolerance from the curve and within tolerance with respect to a possible smooth connection to adjacent curves. The user inspects the solution and accepts or rejects. The control is generally needed since the adjacent curves may also be modified.

3.1.2.11. Relatively narrow neighbouring patches: G-SU-RN. The healer may automatically provide several alternative solutions, including extension of patch 1 to make patch 2 redundant (see the example figure in section 3.1.2.11). It may include an effort to automatically connect the enlarged patch 1 with an eventual patch 3 on the opposite side of patch 2. Another alternative is splitting patch 1 into two parts at a position given by the user, and automatically replacing the part adjacent to patch 2 and patch2 with a new patch 2 that connects with the desired degree of smoothness to surrounding patches, if any. The user makes the decision of which one of the exhibited solutions to take, and possibly enhance further.

Comments on (d)

3.1.2.1. Large patch gap (G_0 discontinuity): G-SU-LG. And 3.1.2.10. Narrow surface or patch: G-SU-NA. Too many possible solutions exist to predict and provide automatic solutions for. The user must take control of how to proceed by indicating what is allowed to be changed and what is not. For instance, one of the surfaces may not be changed at all and the other surface must accommodate for the entire change.

The case when all is allowed to change up to a position tolerance and smoothness tolerance to surrounding surfaces enables an easy recreation without the encountered deficiencies. In the case of styling-dependent parts, such a solution may be out of scope.

What a healer can provide are tools to enable easier fulfilment of the kind of solution that the user has chosen than those available in the native CAD system.

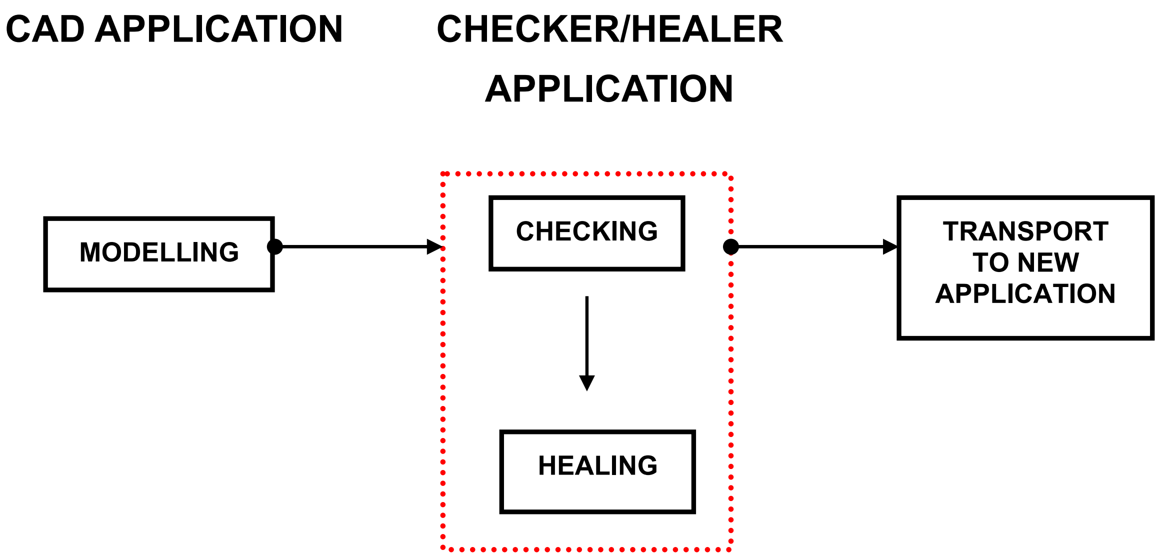


Figure 5. Current healer workflow

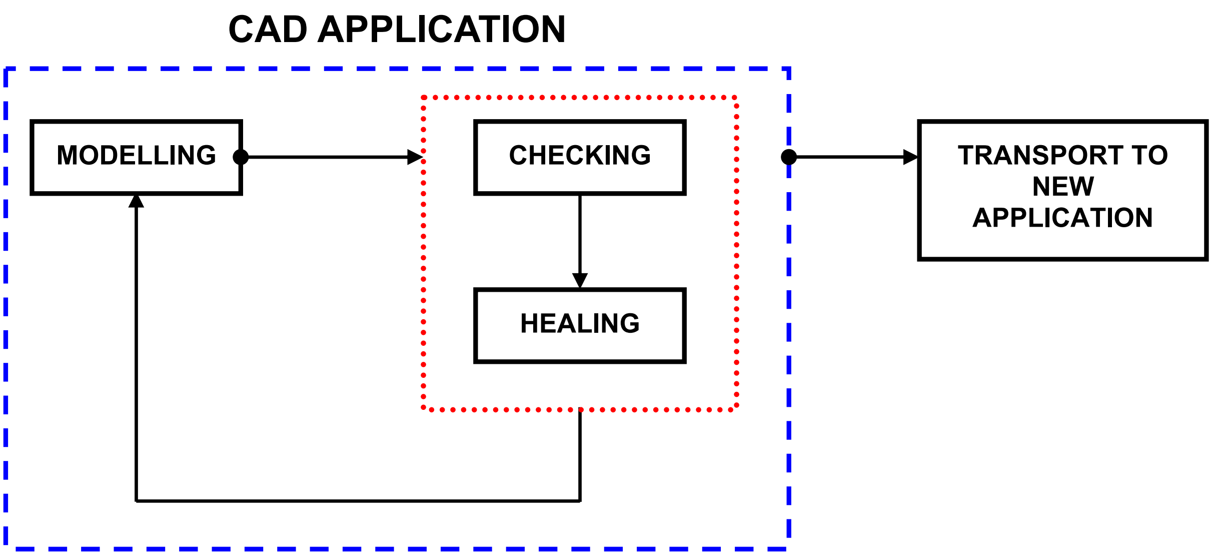


Figure 6. Future healer workflow

(Blank page)

ATTACHMENTS

Attachment A - Glossary

Table 12. Words and phrases

Word/Phrase	Description
	A
Accuracy (of CAD system)	The mathematical value that determines geometric equivalence such as whether two points coincide, two edges coincide, or whether an edge is on a surface. Accuracy is generally dependent on the software package in use, such as a CAD system.
Angle	The difference in direction between two lines, a line and a surface, or two surfaces. In CAD systems, angle values are typically measured between -180 and +180 degrees as follows: if tangent, the angle approaches 0.0 deg; if forming an internal corner, angle is less than 0; if forming an external corner, the angle is greater than 0. Thus a Sharp Face Angle forming a knife-edge would approach +180 degrees and a Sharp Face Angle forming a crack would approach -180 degrees.
Attribute	・A property of an object or ・a named value or relationship that exists for some or all instances of an entity
	B
B-Spline	Basis Spline. A mathematical representation of a portion of a curve, using specific polynomials. A series of B-splines can be combined into complex curves.
Boundary representation (B-rep)	Geometrical representation with a consistent topological structure and explicitly defined geometry for faces and edges. The B-rep model is represented by one or more B-rep solids. Each B-rep solid consists of a single closed shell, which represents its exterior, and an arbitrary number of closed shells corresponding to internal voids. Each shell consists of a collection of faces. Each face is defined by a surface and by loops of edges bounding the surface. The edges meet at vertices and have their geometry defined by curves. The geometry of the curves and surfaces can be complex but is required to be directly defined.
Bounding box	A rectangular box with its sides parallel to the global co-ordinate system planes and meeting the outermost contact point on the part. When all 6 sides are defined, they form a box around the part.
	C
Composite surface	A connected two-manifold, which divides three-dimensional space into two connected components. One of the divided spaces is specified as finite.
Composite curve	A collection of curves whose end points are connected. Individual segments of the curve itself are defined as composite_curve_segment. Parameterisation of a composite_curve is the total parameter range of referred bounded_curve.
Conical surface	A surface of a right circular cone.
Connected	Joined or fastened together.
Connected component	A set of maximal connected portions of a domain.
CSG Model	One type of geometric model. A solid is defined as a result of a series of normalised Boolean operations for a solid model. (CSG=Constructive solid geometry)
Curvature	The ratio of the change in the angle of a tangent that moves over a given arc to the length of the arc.
Curve	An image of a continuous function defined on a set of connected portions of a real line (R^1) in two-dimensional or three-dimensional space, a set of mathematical points that are not independent points.
Cycle	A chain in which vertexes and edges in a graph appear alternately, with the first vertex and the last vertex being identical.

D	
Degenerate Surface	Surface possesses one or more critical points where the normal vector can not be calculated. In a typical degenerate surface situation two adjacent boundaries are parallel or collinear at one shared end point, or the length of one boundary is zero.
Degree (of a curve or surface)	The maximum polynomial degree of the underlying mathematical representation of a curve or a surface.
Dimensional tolerance	Specification of the maximum and minimum acceptable dimensions for an acceptable dimensional range as opposed to a specified nominal dimension for machining some area.
Domain	A set of mathematical points inside a model space, which corresponds to an entity.
E	
Edge	An edge is a topological element that corresponds to a connection between two vertices.
Extent	Degree of measured content of an entity domain. This measured degree is expressed in units corresponding to the dimensionality of the entity. That is to say, for dimensionalities 1, 2 and 3 the extents are respectively length, area and cubic volume.
F	
Face	A topological entity that corresponds to the portion of a surface bounded by a loop (bounded-surface).
Features	A portion of an object that comprises geometric or other characteristics and is used for some specific functional purpose. The function may be related to design, manufacturing or use.
Fillet	A rounded shape used to smoothly connect two adjacent faces.
Filter	A process of reducing the amount of information that is required to complete the task.
Fixture	A piece of equipment that holds parts or assemblies in the correct position for manufacturing operations.
G	
G_0	When the end point, or boundary, of one element is coincident with the end point or boundary of a neighbouring element, they are characterised as having G_0 continuity (geometric position continuity).
G_1	When the normals of neighbouring elements with positional continuity are also coincident, they are characterised as having G_1 continuity (geometric tangential continuity).
G_2	When the center points of curvature of neighbouring elements, measured in normal sections, with positional and tangential continuity are coincident, they are characterised as having G_2 continuity (geometric curvature continuity).
Gap	Distance between two geometric entities that are topologically defined to be coincident.
Geometrical tolerance	The allowable difference or deviation from a given position within a part or shape.
H	
Healing	A (automated) correction to be performed by an application or software on one or more objects of a CAD model where a PDQ criterion violation was detected.
I	
Inconsistent (geometry)	Orientation of geometrical or topological element is not compliant with corresponding elements.
Inside	The inner side of a geometry.
Iso-parametric curve	A curve on a surface where one parameter is fixed (constant)and the other parameter is running in its definition interval.
K	
Knot	A specific parameter value that is used when defining the basis polynomial function of a spline element.

L	
Layer	A method of organizing elements in CAD systems such as for visualisation purposes.
Loop	A topological entity constituted of connected edges.
M	
Meta-data	Information that describes characteristics of data sets.
N	
Non-manifold	The misuse of a topological entity such as an edge used by more than two faces.
NURBS(Non-Uniform Rational B-Spline)	A curve or surface expression in which each segment is a B-Spline that is aligned in a rational manner and using basic polynomial splines with non-uniform knot distribution. NURBS can define a circle and a sphere precisely. Segment or patch is a section that is divided by segment values.
Normal	Being at right angles; perpendicular to a surface or plane.
Numerical Control (NC)	A manufacturing method which controls the movement of a machine tool with numerical values and servomechanism.
O	
Offset (entity)	A curve or surface that maintains a constant distance from a given curve or surface.
Overall extent	Length of diagonal of bounding box around entity.
Overlap	A condition in which two entities share part of a shell, a face, a surface, an edge, or a curve.
P	
Parameter	A variable that constrains or controls some aspect of an object.
Parametric Modelling	A method of modelling geometry or other characteristics by defining relationships or values among parameters such that variations can be easily generated.
Part	A component of a vehicle/machine/equipment (in CAD/CAM/CAE models).
Patch	The portion of a surface formed by a single polynomial expression (depending on two surface parameters). Refer to NURBS.
Planar curve	A curve defined on a plane.
Point marker symbol	Symbol to represent the point (e.g., star, dot).
Polynomial (entity)	A geometric element defined by a polynomial expression.
Presentation	Various means of expressing or displaying aspects of an object (e.g., shaded model, drawing, bill of material).
Proximity	Distance between two geometric entities that are topologically defined to be separate.
R	
Region	Portion of a face bounded on two or more sides by portions of outer and/or inner edge loops.
S	
Segment	A section of curve represented by a single polynomial expression. Refer to NURBS.
Shell	Set of sewn faces. An open shell can be used to model a sheet metal part. A closed shell defines the outer boundary of a solid.
Solid	Complete representation of a shape in which the points of its interior are all connected. Any point in space can be classified with regard to a given solid as being inside, outside, or on the boundary. It is possible to have a solid with two or more distinct volumes.
Spline	A curve defined as a sequence of curve segments, which are connected so that certain conditions of continuity are met.
Surface	A geometric element defined by mapping a 2D parametric square into 3D Euclidean space by using specific mapping functions.

T	
Tolerance	The permissible deviation from a specified value. See Geometrical tolerance or Dimensional tolerance.
Tooling	Equipment used to shape a part through direct contact, such as moulds, dies, and cutting tools.
Topology	The logical and positional relationship of a set of geometric entities that comprises a geometric structure.
V	
Vertex	The topological construct that represents the intersection of two or more edges.
Visualisation	A form of representation to view an object's geometry or other characteristics.
Volume	A bounded portion of space having length, width and thickness.

Table 13. Acronyms and abbreviations

Acronym/ abbreviation	Full Spelling	Description
AIAG	Automotive Industry Action Group	Organisation to promote standardisation in automotive industries of the United States.
API	Application programming interface	Library of commands to access the capabilities of a system, or data within a software package.
B-Rep	Boundary Representation	See Glossary.
CAD	Computer-aided design	Use of computer-based methods to design an object.
CAE	Computer-aided engineering	Use of computer-based methods for various engineering purposes.
CAM	Computer-aided manufacturing	Use of computer-based methods for the preparation and simulation of machining.
CFD	Computational fluid dynamics	Simulation method of behaviour of fluids in space.
CMM	Co-ordinate measuring machine	Device to determine points in space.
FCAI	Federal Chamber of Automotive Industries	Association of automotive industries in Australia.
FDM	Fusion discharge machining	Method for rapid prototyping of metal parts.
FEA	Finite element analysis	Simulation method of physical properties under specific conditions.
GALIA	Groupement pour l'Amelioration des Liaisons dans l'Industrie Automobile	Association of automotive industries in France.
GOSET		French STEP expertise centre.
IGES	Initial Graphics Exchange Specification	A type of intermediate file format that is used to exchange data among different CAD systems. The standard of figure data established by the American National Standards Institute (ANSI).
JAMA	Japan Automobile Manufacturers Association, Inc	Japan Automobile Manufacturers Association Inc.
JAPIA	Japan Automobile Parts Industries Association, Inc.	Japan Automobile Parts Industries Association Inc.
LOM	Laminated object modelling	Method for rapid prototyping using sheets of paper, wood, plastic or metal.
ODETTE	Organisation for Data Exchange by Tele Transmission in Europe	Organisation formed by automotive industries in Europe in order to improve international competitiveness.
PDM	Product data management	Software to store and manage product data and meta-data.
ProSTEP		The generic term for organisations to promote STEP in Germany. It consists of the ProSTEP iViP Association (non-profit organisation) and the PROSTEP AG (corporation).
SASIG	Strategic Automotive product data Standards Industry Group	Organisation formed by associations of automotive industries in Japan, the United States and Europe to promote the standardisation and utilisation of product data creation and handling methods.

Acronym/ abbreviation	Full Spelling	Description
STEP	Standards for the Exchange of Product model data	Common name for the standards regarding representation and exchange of product models, which are being developed by ISO (International Organisation for Standardisation) 10303. Product data exchange format for which International Organisation for Standardisation (ISO) is promoting the standardisation. It can be used to exchange not only shape data such as wire frame models, surface models, and solid models but also product composition data exchange control information and others.
STL	Stereolithographic Language	Data format primarily used for rapid prototyping.
VDA	Verband der Automobilindustrie	Association of automotive industries in Germany.
VDA4955-V2	VDA recommendation 4955 Version 2	VDA prepared PDQ guidebook titled "Scope and Quality of CAD/CAM Data" and is advancing the improvement of model quality.
VDAFS	VDA Flächenschnittstelle	Intermediate format for Data exchange specified by the Association of Automotive Industries of Germany.
VDMA	Verband Deutscher Maschinen- und Anlagenbau	Organisation of machine, tool and die industries in Germany.

Attachment B - Mapping Between Element Types

Table 14. Mapping between element types

Element Type	STEP	IGES	CATIA V4	CATIA V5	UG	CADCEUS	I-DEAS	Pro/E
Point	Cartesian point	116	PT	Point	Point	Point	Point	Point
Curves								
B-Spline curve	b_spline_curve	126	CRV	Curve, 3DCurve, Spline	b-curve	B-Spline curve	curve	B-Spline curve
Polynomial curve	-	112	CRV	curve	-	-	curve	(spline) curve
Line	line	110	LN	Line	Line	Line	curve	(line) curve
Circle	circle	100	CIR	Circle	arc, circle	arc, circle	curve	(arc) curve
Ellipse	ellipse	104	ELL	Ellipse	ellipse	ellipse	curve	(conic B-spline) curve
Hyperbola	hyperbola	104	HYP	Hyperbola	hyperbola	-	curve	(conic B-spline) curve
Parabola	parabola	104	PAR	Parabola	parabola	-	curve	(conic B-spline) curve
Composite curve	composite_curve	102	CCV	wire	-	composite curve	WF Loop	composite curve
Group of joined lines	polyline	106/ 12	CRV	Polyline (internal wire)	-	-	curve	composite curve
Transformed curve	curve_replica	-	-	-	-	-	-	
Trimmed curve	trimmed_curve	-	CRV	Trimmed curve	trimmed curve	Curve	-	Curve
Offset curve	offset_curve	130	CRV	3D curve offset	-	-	curve	Curve
UV-curve	pcurve	-	-	pcurve	-	-	-	Curve
Surface curve	surface_curve	142/ 141	-	intersct.、projection, boundary	-	-	Loop	-
Surfaces								-
B-Spline surface	b_spline_ surface	128	SUR	Surface	b-surface	B-spline surface	Surface	(Nurbs) Surface
Polynomial surface	-	114	SUR	Surface	-	-	Surface	(Spline) Surface
Cone surface	conical_ surface	-194	SUR	-	cone	cone	Surface	(Cone) Surface

Element Type	STEP	IGES	CATIA V4	CATIA V5	UG	CADCEUS	I-DEAS	Pro/E
Cylinder surface	cylindrical_ surface	-192	SUR	cylinder	cylinder	cylinder	Surface	(cylinder) Surface
Offset surface	offset_surface	140	SUR	Offset surface	offset surface	-	Surface	Surface
Ruled surface	-	118	SUR		-	-	Surface	(Ruled) Surface
Plane	plane	108	PLN	plane	plane	Plane	Surface	(datum) Plane
Composite surface	rectangular_ composite surface	-	-	Join	-	composite surface	-	-
Cut out of a surface	rectangular_ trimmed_ surface	-	SUR		-	-	-	-
Extruded surface	surface_of_ linear_ extrusion	122	SUR	Tabulated cylinder	extruded surface	translational sweep surf.	Surface	(Tabulated cylinder)
Sphere surface	spherical_ surface	-196	SUR	sphere	sphere	sphere	Surface	-
Rotation surface	surface_of_ revolution	120	SUR	revolution	surface of revolution	rotational surface	Surface	Surface(of revolution)
Torus surface	toroidal_ surface	-198	SUR	-	torus	torus	Surface	Surface (torus)
Trim Geometry								
Face	curve_ bounded_ surface	144/ 143	FAC	Face	-	Surface	Face	Surface
Boundary	boundary_ curve	142/ 141	CRV	Boundary	-	boundary	Loop	Contour (edge loop)
Topological Elements								
Open shell	open_shell	-	SKI	shell	shell	open composite surface	-	Quilt
Closed shell	closed_shell	-514	VOL	Volume	shell	closed comp. surface	Shell	Quilt
Solid Elements (Brep)						-		
Vertex	vertex_point	-502	-	vertex (internal entity)	vertex	vertex	Vertex	(vertex)
Loop	vertex_loop	-508	-		loop	vertex loop	Loop	-
Edge	oriented_edge + edge_curve	(504) (508)	-	edge	edge	Edge	Loop	Edge
Edge loop	edge_loop	-508	-	-	loop	edge loop	Loop	Contour (edge loop)
Boundary	face_bound	(in 510)	-	Boundary	loop	boundary	Face	Contour (edge loop)
Trimmed surface	advanced_face / face_surface	-510	-	face	face	surface	Face	Surface
Solid	manifold_ solid_brep	-186	SOE	Solid	body	body	Region	protrusion

(Blank page)

Table 15. Mapping between Non-Geometrical element types

Entity Name	STEP	IGES	CATIA V4	CATIA V5	UG	CADCEUS	I-DEAS	Pro/E
Startup Environment			Project environment, Declaration File ENV file, Startup model	Settings & Standards (XML), Environment file	ugii_env.dat (Windows). ugii_env (Unix)	Startup file	Param file	Start model
Accuracy Parameter			Model dimension	intern Resolution	Distance Tolerance, Angle Tolerance	Distance Tolerance, Angle Tolerance	(fix)	Accuracy
Hybrid model			Hybrid model	Hybrid Part	—	—	—	
Multi-Solid model			Multi-Solid model	Multi-Solid model		—	Workbe nch	
Item			Model, Session	CATProduct, CATPart CATDrawing	Part, Assembly, Drawing,	Work space, Part object, Sheet object		Part,Assembly, Drawing
Physical file name (always on OS level)			Model name	CATPart name, CATProduct name, CATDrawing name	Part file name	—	Modelna me	Model name
Item property			FILLE-COMMENT, UDB	Property-Description	Property, Part attribute	Object attribute	Property	Attribute, Parameter
Item data consistency			CATCLN	Healing assistance, CATDUA	VDA checker/Che ck-Mate/Part cleanup	Consistence check	Diagnos tic-Part	Geometry check /Model check
Reference set			-	-	Reference Set	—		--
Encapsulated Entity			Detail, Symbol	2D-Detail, PowerCopy, UDF	-	—	Feature Copy, UDF	-
External item reference			Import elements	Reference Links, CCP Links, Import Links,view-link	WAVE-Links, Interpart Expressions	—	Excel	External copy
Simplified part			Light model	CGR	Simplified representatio n, faceted body	—	CWA-Part	ShrinkWrap
Bounding Box			Workspace	—		—	Boundin g Box	--
Group			Set	—	Group	Group	Group	Group
Layer			Layer	Layer	Layer	Layer	-nur bei 2D	Layer
Instance			DITTO	Instance		Instance	Instance	Instance
Layer group			Filter	Display filter	Layer category	—	Filter	--
co-ordinate system			AXIS	Coordinate system	(Absolute/ Work)	Coordinate system	Coordin ate	Coordinate system

Entity Name	STEP	IGES	CATIA V4	CATIA V5	UG	CADCEUS	I-DEAS	Pro/E
					Cordinate system CSYS		system	
Units			Model Units	Units	Units	Units	Units	Units
Scale			Model Scale	Scale	Scale	Scale	Scale	Scale
Transformation			Transformation	-	-	—	Transfor mation	—
Assembly (relationship)			Session	CAT Product	Assembly	Assembly	Assembl y	Assembly
Assembly constraint			-	Constraint	Mating condition	Assembly placement conditions	Assembl y constrai nts	Constraint
Form feature			Feature	Feature	Form feature	—	Feature	Feature
Unresolved feature			Unresolved Feature	Unresolved feature	Inactive Feature, Unresolved feature	—	—	Incomplete Feature
Inactive Feature			Inactive Feature	Inactive Feature	Restraint form feature, Suppressed feature	—	supresse d feature	Suppressed Feature
Element name			IDENTIFIER	Feature name	Body property	External name	Entity	Item name
prohibited element						—		
User defined element			User defined element	UDF	Knowledge Fusion	—	UDF	UDF
Color (settings)			COLOR	Color	Color	Color	Color	Color
Point marker symbol			POINT TYPE	Point symbol	Point/Point line	Point type	Referen zpoint	Point type
Line type			LINE TYPE	Line type	Line font	Line type	Line Style	Line type
Line width			THICKNESS	Line width	Width	Line thickness	Linewid th	Line thickness
Element visibility			Show/Noshow, Pick/NoPick	Show/Hide, Pick/NoPick	Blank	Display on/Display off	Show/N oshow,H ide	Hide/Unhide
Display mode			SHD(Shadng)	Display mode	Display mode	—	Wirefra me,shad eing,mix ed mode	Display mode
Sketch			Sketch	Sketch	Sketch	Sketch	Sketch	Sketch

Table 16. Mapping between Drawing entity types

Entity Name	STEP	IGES	CATIA V4	CATIA V5	UG	CADCEUS	I-DEAS	Pro/E
ISO conformable text				JIS, ANSI, ASIM, ASME_3D	—	—	—	--
CAD source notice				Link information	Master model	—	Ownership	CAD source notice
Reference on external database or library			Library Detail	Detail Link	Reference on library	—		Reference on external object
2D Drawing			DRAW	CATDrawing	Drawing	Drawing Object	Drawing	Drawing
2D/3D linkage				2D/3D linkage		—		
Drawing sheet			DRAFT	Sheet	Drawing sheet	Sheet object	Multi sheet	Sheet
Plot frame			Plot Window	print area	Choice of the plot geometry	—	—	Plot frame
Size of view frames			AUXVIEW/FRAME/SIZE, Drawing Frame	Frame and Title Block	View boundary of the drawing member	—	Main View	View boundary
VIEW			VIEW	View	View	View	VIEW/Model View	View
Fake Dimension			Fake Dimension	Fake Dimension	Dimension has an editorial text	Edited dimension	Out of scale	Overwritten dimension
display accuracy of dimensions			Dimension Tolerance	Dimension Tolerance	Dimension Tolerance	Precision of fractional part	—	Dimension tolerance
associative dimension				Associative dimenson		—		Associative dimenson
View dependent object			Element display via /SOLID/MODIFY /DRESS UP	Element display via Component Properties Drafting	View dependent object	—		View associated draft geometry
View projection method				First Angle / Third Angle		—		View projection method

Attachment C - Recommended Values

The Recommended Values table has been removed from the document.

Please refer to national addendums for appropriate recommendations.

Attachment D - Formsheet

The SASIG CAD/CAM Data Quality Exchange Agreement document is foreseen to support a Data Exchange Process Model starting with an agreement regarding CAD data creation and exchange details, followed by the regular exchange of CAD data including the documentation of its quality. Some kind of feedback may complete the process and confirm the usability of the data on the receiver's side.

The Agreement document is intended to allow for flexibility in its use.
The example at the end of this attachment is provided to suggest how member organisations and users can establish appropriate exchange agreements. It can, by checking or filling in appropriate fields, be used for any situation, from an entire corporation to a single part. Fields are provided for the agreeing parties to tailor the Agreement to meet all situations.

It is recommended that, if a wide-ranging Agreement is made, such as a Corporate / Master Agreement, any organisation(s) within the agreeing parties that finds a need to alter from the Master Agreement should publish its specific agreement with notation in the Special Conditions referring to the Date of the Master Agreement.

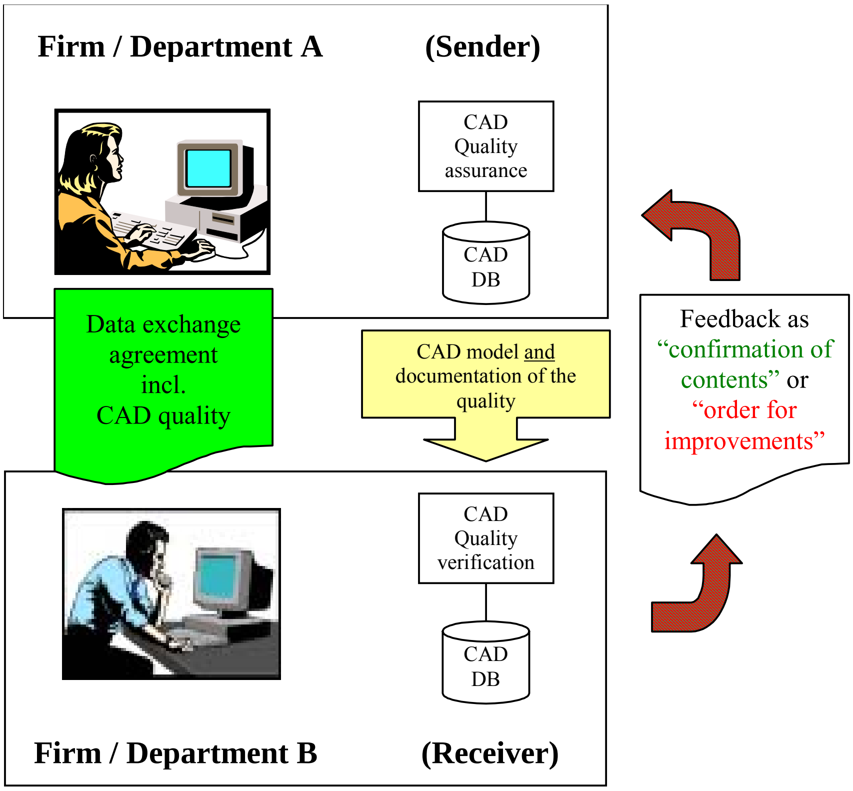


Figure 7. Data exchange process model

Procedural instructions:

1. Start by stipulating the project partner(s), (sender(s) and recipient(s) who will be responsible for the quality of the product data as well as the area of validity covered by the agreement (project/component part). Together with the date of agreement, this information shall represent explicit "fundamental specifications" for multi-sided agreements (one to one, one to many, many to many, supply chain inclusion as necessary). Define responsibility, validity, partners, a unique and clear framework.
2. Stipulate the intended use of the product data that are to be supplied. Often, several intended uses must be stipulated and where possible, listed in chronological order (relative to the product development process, i.e., concept, design, analysis, review, release, prototype, manufacture, etc.). Only then is it meaningful to stipulate the associated quality criteria on one form. However, it can also be more understandable to employ one form per intended use. At this point, one can use the categories (criteria and characteristics) according to update.
3. Classify each intended use of a model/geometry type, where required, the model type and as well as additional stipulations (e.g., for the degree of detailing).
4. In the left hand column, mark the pertinent quality criteria for the respective intended use (discipline) which the sender and recipient have jointly assessed to be required. Please do not mark all criteria as required but concentrate on the really important "minimum" requirements. A detailed description of the criteria may be found in

SECTION II. **Note:** There may be more than one form necessary per/discipline, depending on the scope of the project.

5. Determine, then agree and adopt the limiting values for the “promoted” criteria. Use recommended values in Attachment C, or use your own. All parties need to be considered, included, and in agreement. You should fill out only necessary criteria. You may have multiple contracts between different partners in the project or throughout the supply chain as necessary.
6. CAD data could be viewed as "work in progress" and not complete. This should be considered when defining the agreement between parties.

CAD/CAM Data Quality Exchange Agreement (Example) Date: ____ / ____ / ____

Between:

______________ ______________ ______________ ______________
Company Name Department Contact Name Contact Phone No.

______________ ______________ ______________ ______________
Company Name Department Contact Name Contact Phone No.

Applies to:

______________ ______________ ______________ ______________ ______________ ______________
Corporate / Master Car Program Body / Model Design Discipline Part or Assembly # Version

By this Agreement, the companies agree that all CAD and/or CAM data to be exchanged will conform to this agreement and will either meet or exceed the Recommended, or Specific, values for all fields / criteria checked as required.

Accuracy/Restrictions: CAD System: ________ HW Platform(s): ________ Model Dimension: ________

CAD-Model type: ❑ Wireframe ❑ Surface ❑ Solid ❑ Space Claim (Volume only) ❑ ____________

Drawings: ❑ NONE ❑ Reduced ❑ 2D only ❑ Associated to 3D model

Intended Model Use: ❑ Bid/Offer ❑ Prototype ❑ Background Info ❑ DMU ❑ Release ❑ ________

Max. Model Size: ______ MB ❑ Spec. Conditions: ______________________ ❑ see Attach.

Conformance Required		Code	Criteria Identification	Example (CATIA) Value (mark if …)	Specific Value	Checked	Due Date or Milestone
Model	☐	MU	Multi-Solid Model	Count > 1	____	☐	____
	☐	HY	Hybrid Model	Yes	____	☐	____
Geometry	☐	LG	Large Gap in Element	> 0.02 mm	____	☐	____
	☐	NT	Non Tangent Segments	Angle > 1.0 deg	____	☐	____
	☐	NS	Non Smooth Segments	> 0.8 < Ratio < 1.2	____	☐	____
	☐	GP	Edge Gap in Faces	Gap > 0.02 mm	____	☐	____
	☐	VF	Vertex Gap in Faces	Gap > 0.02 mm	____	☐	____
	☐	CR	Curvature Radius of El.	Radius < 0.1 mm	____	☐	____
	☐	WV	Waviness of Elements	Yes	____	☐	____
	☐	FO	Folded Elements	Angle = 120 deg	____	☐	____
	☐	DC	Degenerated Curve of El.	Length < 0.1 mm	____	☐	____
	☐	DP	Degenerated at Point	no Normal defined	____	☐	____
	☐	SA	Sharp Angle of Elements	Angle >178 deg	____	☐	____
	☐	TI	Tiny Elements	Extent < 0.1 mm	____	☐	____
	☐	NA	Narrow Area	Width < 0.02 mm	____	☐	____
	☐	RN	Relatively Narrow El.	Ratio u/u,v/v > 100	____	☐	____
	☐	IS	Intersection	Proximity < 0.01 mm	____	☐	____
	☐	AN	Analytical Elements	Approximation fails	____	☐	____
	☐	IK	Indistinct Knots	Distance < 0.001	____	☐	____
	☐	HD	High Degree Elements	Degree > 9	____	☐	____
	☐	FG	Fragmented Elements	> 200	____	☐	____
	☐	CL	Closed Elements	Distance start /end < 0.01 mm	____	☐	____
	☐	IT	Inconsistent Topology	opposite orientation	____	☐	____
	☐	FR	Free Edge	not sewn edges	____	☐	____
	☐	NM	Non-Manifold (OU Edge)	> 2 faces per edge	____	☐	____
	☐	OU	Over-Used Vertex	> 6 edges per vertex	____	☐	____
	☐	MU	Multiple Elements	> one element	____	☐	____
	☐	RE	Embedded Elements	within < 0.02 mm	____	☐	____
	☐	UN	Unused Patches	Yes	____	☐	____
	☐	VO	Solid Void	Yes	____	☐	____
	☐	NU	Non-updatable Solid	Yes	____	☐	____
	☐	MH	Missing Solid History	Yes	____	☐	____
	☐	UH	Unused History	Yes		☐	

Attachment E - SASIG-ODETTE Cross-Reference

Much of Section 3 of this document was borrowed from the ODETTE CAD data quality document QAM V2, which was, in turn, built from the VDA4955 V2. Table 17 provides a criterion-by-criterion cross-reference between the ODETTE QAM V2 document and this SASIG PDQ document where the content is the same. This document (SASIG PDQ) also contains additional content, including criteria, beyond the ODETTE and VDA documents. Cells with the symbol N/I (not included) are not covered as criteria in this document, though the sense of the content may be found elsewhere in the guideline.

Table 17. SASIG-ODETTE Cross Reference

ODETTE QAM V2 Item title	Criteria Encoding	
	QAM V2	SASIG-PDQ V1.1
Wireframe geometry		
Tiny elements, tiny segments	M1	G-CU-TI
Identical elements (by approximation)	M2	G-CU-EM
Continuity	M3a M3b M3c	G-CU-LG G-CU-NT G-CU-NS
Polynomial degree	M4	G-CU-HD
Waviness in a planar curve	M5	G-CU-WV
Self-penetration / intersection	C7	G-CU-IS
Multiple knots	M6	G-CU-IK
Surface		
Tiny elements, tiny segments	M1	G-SU-NA G-SU-RN G-SU-TI
Tiny segment edge	SU8	G-SU-DC
Identical elements (by approximation)	M2	G-SU-EM
Continuity	M3a M3b M3c	G-SU-LG G-SU-NT G-SU-NS
Polynomial degree	M4	G-SU-HD
Waviness	M5	G-SU-WV
Minimum curvature radius	SU9	G-SU-CR
Angle between the boundary curves of surfaces	SU10	G-SU-DP
Reversal of normals	SU11	G-SU-FO
Patch numbers / partitioning	SU12	G-SU-FG
Unoccupied patch rows	SU13	G-SU-UN
Multiple knots	M6	G-SU-IK
Bounded surface (face)		
Tiny elements	M1	G-FA-NA G-FA-RN G-FA-TI
Tiny edge curves	M1	G-ED-TI
Continuity of boundary curves	M3a	G-LO-LG
Penetration or contact of boundary curves	F14	G-LO-IS G-FA-IS
Proximity of boundary curve to surface	F15	G-FA-EG
Parallel path within a boundary curve	F16	G-LO-IT
(Proportional) Number of segments in boundary curves	F17	G-ED-FE

Composite surface, topology		
Continuity	M3a M3b M3c	G-SH-LG G-SH-NT G-SH-NS
Junctions	T18	G-SH-NM ~~3.2.6.8~~
Alignment / Orientation of similar normals	T19	G-SH-IT
Knife edges	T20	G-SH-SA
B-rep solids		
General criteria for B-Rep solids	M1-T20 w/out C7	N/I
Tiny elements	M1	G-SO-TI
Distance Vertex - Edge	SO21	G-FA-VG
Distance Vertex - Face	SO22	G-FA-VG
Drawing elements		
Tiny elements	M1	G-DW-TI
(By approximation) Identical elements	M2	G-DW-EM
IGES conformable texts	D28	G-DW-CT
Polynomial degree	M4	N/I
Organisational data quality		
Administrative information / Model name	2.4.1	G-MS-AI
Model structuring – Co-ordinate system(s)	2.4.2.1	3.1.2
Model structuring – Changes	2.4.2.2	3.1.1
Drawings		
Views	2.4.3.1	G-DW-VW
Referencing of 3D models in drawings	2.4.3.2	G-DW-MM
CAD source notice	2.4.3.3	G-DW-GR
References on external databases and libraries	2.4.3.4	G-DW-ER
Solids		
Capability for regeneration	2.4.4.1	G-SO-NU
Derivation of other geometrical forms	2.4.4.2	N/I
Derivation of other geometrical forms – Edges/Individual faces	2.4.4.2.1	G-ED-AN G-FA-AN
Derivation of other geometrical forms – Surface groups	2.4.4.2.2	3.1.5
Preference for canonical elements	2.4.4.3.1	3.1.6
Preference for form features	2.4.4.3.2	G-FE-FF
Preference for solid functions by hybrid models	2.4.4.3.3	N/I
No by-passing of the history	SO23	G-SO-MH 3.2.7.3
No utilisation of redundant (auxiliary) geometry	SO24	3.1.4 G-SO-UH
No (unintentional) cavities	SO25	G-SO-VO
Systematic development	2.4.4.3.7	3.1
Reduction of the degree of detailing (features)	2.4.4.3.8	G-FE-RD
Multi-body solids	SO26	G-SO-MU
Multi-solid parts	SO27	G-MO-MU ~~3.2.8.2~~

Attachment F - BUSINESS CASE

This Attachment provides templates to determine the cost of product data quality problems. These tools and methods will help the user generate a business case for improving product data quality.

F Building a Business Case for Product Data Quality Improvement

No company should undertake a change in the way it does business without determining the likely financial benefit of the change. This part of the guidelines is designed to assist in working out the likely benefit to a company from implementing product data quality (PDQ) systems.

The following sections provide templates that will help build a business case for improving PDQ. These templates are intended to be a guide, not a perfect solution, because individual companies have their own special circumstances. The templates do not try to formally assess the return on investment, largely because that is very difficult to do for the kinds of change addressed here. However, they do provide a useful starting point for such a calculation. Each template also provides a place to capture unique cost factors beyond the ones specifically raised on the template. In that manner, a user can customise each template to a particular situation.

As much as possible, the costs are collected in terms that make sense to a person conversant in the particular area of interest. For example, costs are collected in terms of monthly numbers, by component, and by analysis. When data are entered into the templates in those terms, they are then aggregated to an annual cost level. Direct entry of annual numbers is primarily used for the open-ended items involving costs not directly addressed by the templates.

In the templates, some costs may be difficult to document with any real accuracy. The cost elements listed in these templates are, however, real and potentially significant. The most useful results will come from the best estimate of such costs. The best way to fill out the templates is to have different people fill out the different templates, according to their areas of expertise.

For some costs, it may be difficult to estimate the level of effort because the problem is a rare occurrence but significant when it does happen. An example of this might be where a problem was not detected until tooling was made. This would be expensive to fix but rarely happens in most companies. For this kind of situation, the cost can be estimated by multiplying an average cost for tooling by an estimate of the probability that such a situation will occur.

Filling in the templates allows focusing on the major costs that affect a particular company. Only elements of cost directly related to product data quality should be considered in filling out these templates. The following pages present the detailed templates that should be used to calculate the costs of product data quality problems in a number of areas. These templates include:

1. CAD-related PDQ Costs (Annual)
2. CAM-related PDQ Costs (Annual)
3. CAE-related PDQ Costs (Annual)
4. Prototyping PDQ Costs (Annual)
5. Digital Mock-up PDQ Costs (Annual)
6. PDM-related PDQ Costs (Annual)

In addition, there are two other special-purpose templates:

7. Start-up and Annual PDQ Improvement Implementation Costs
8. Net PDQ Improvement Savings for First and Subsequent Years

Template 7 is for determining the start-up and annual costs of undertaking a PDQ improvement process. This template applies to any improvement process, so there is only one presented here. Template 8 is a summary template that collects the costs from the various product data quality templates (1 through 6) and Template 7 in one place.

There are other areas that might make sense for templates. In particular, the exchange of data between product data management (PDM) systems will eventually become an important issue. Unfortunately, this is such a new concept that there is not enough experience to suggest how to capture costs from problems with it. If other activities receive CAD models on a regular basis outside of the ones covered by templates 1 through 5, new templates can be constructed from the examples provided here.

F.1.1 Annual CAD-Related Costs of Product Data Quality

The following are line-by-line instructions for completing the template F.1.7 for the costs of product data quality problems within and between CAD systems. These costs explicitly do not include costs that occur in other activities. Those other costs are addressed in the other templates. The line-by-line instructions describe what data go in the various places in the template. Calculated entries are described directly on the template rather than here.

F.1.2 Supporting information

Line 1 – Cost of labour

This is the cost rate per hour per employee. This includes employee's pay-rate, benefits, equipment costs and overhead. This value is used in some of the later computations.

F.1.3 Costs of corrupted data prior to exchange or translation

The calculations in this section are primarily based on monthly numbers. They apply to work done on a file before any exchange has taken place.

Line 2 – Number of CAD files or models worked on per month

This is the average number of CAD files or CAD models created, modified, updated, exchanged, reviewed, etc. in a given month. It is used as a basis for comparison.

Line 3 – Average time spent checking for corrupt data per file or model

This is the amount of time that is spent checking a file or model for file contamination or other problems. The results of this check normally will dictate whether the file or model is usable or salvageable.

Line 5 – Number of corrupted files or models

This is the average monthly number of CAD files or models that have become corrupted during the modelling phase, i.e., before any exchange or translation takes place. The problems can be due to bad geometry, poor CAD construction modelling practices, or avoidable weaknesses of the CAD system. The files or models are usually deemed unusable.

Line 6 – Average time spent fixing or modifying data per file or model

This is the amount of time that is spent trying to fix a corrupted file or model. The file or model may be salvageable after it has been modified. Figure the average over the number of corrupted models listed in Line 5, even though some models may have been so obviously bad that no attempt to fix the model was made.

Line 7 – Average time spent on data re-entry or restarting per file or model

This is the average amount of time that is spent re-entering the corrupted geometry. If the original file or model is determined to be corrupted, unusable, or unsalvageable, creating a new file or model may be necessary. Figure the average over the number of corrupted models listed in Line 5, even though some models were fixable and therefore there was no need to redo them.

Line 13 – Costs due to data re-entry, modification, or re-creation errors

These are the average annual costs related to errors or mistakes made during the data re-entry or re-creation process. If data are entered incorrectly, this may cause additional expense. While this number may be difficult to determine or estimate, this category of costs does exist.

Line 14 – Other costs related to corrupt data

These are any additional average annual costs related to corrupt data that may be company-specific or not mentioned on this worksheet.

F.1.4 Costs of failed CAD-to-CAD data exchanges

The calculations in this section are based primarily on monthly numbers. They take into account only the work done after an exchange has taken place.

Line 16 – Number of data exchanges per month (internal and external)

This is the average number of CAD files or models that are exchanged in a given month. A CAD exchange can take place either native between CAD systems or non-native, where a translation may be necessary. Exchanges can be internal to the company; between employees in a department or between departments, facilities, or divisions. External exchanges take place between trading partners. This value is used for comparison.

Line 17 – Number of exchanges with corrupted data

This is the average number of CAD files or models that have failed to complete the exchange process in a given month. This is assuming that the file or model was contaminated *prior to* exchange or translation. The files or models are usually deemed unusable.

Line 18 – Number of exchanges with lack of data

This is the average number of CAD files or models that are found to be incomplete in a given month. This considers only those files or models already incomplete prior to the exchange. The files or models are usually salvageable with modification or data re-entry.

Line 19 – Number of exchanges with too much data

This is the average number of CAD files or models that are found to contain too much detail in a given month. This takes into account only files or models with too much data prior to the exchange. The files or models are usually salvageable with modification or data re-entry.

Line 21 – Checking a file or model for corrupt data after exchange

This is the average amount of labour time that is spent checking a file or model for file contamination after an exchange has taken place. This normally will dictate whether the file or model is usable or salvageable.

Line 22 – Average time spent checking a file or model for insufficient data

This is the average amount of labour time that is spent checking a file or model for completeness of the data. Missing data normally need to be added to the CAD file or model.

Line 23 – Average time spent checking a file or model for excessive data

This is the average amount of labour time that is spent checking a file or model for data that is extraneous to the purpose for which the data are to be used. Excess data often need to be stripped out of the CAD file or model.

Line 26 – Average time spent fixing or modifying data per exchange

This is the average amount of time that is spent trying to fix files or models for which the exchange has failed, either by fixing or removing the excessive or corrupt data. The file or model may be salvageable after it has been modified.

Line 27 – Average time spent on data re-entry or restarting per exchange

This is the average amount of time that is spent re-entering the corrupted or missing geometry. If the exchanged file or model is determined to be corrupted, unusable, or unsalvageable, creating a new file or model may be necessary.

Line 28 – Average time spent reprocessing, re-sending, or receiving data per exchange

This is the average amount of time that is spent to reiterate the exchange process (i.e., not including the original try at sending the data). If the exchanged file or model is determined to be unusable or unsalvageable, repeating the exchange may be necessary. This includes any prep time, processing time for translations, and time spent receiving or re-sending the exchanged CAD data.

Line 34 – Costs due to data re-entry, modification, or re-creation errors

These are the average annual costs related to errors or mistakes made during the data re-entry or re-creation process. If data are entered incorrectly, this may result in additional expense.

Line 35 – Other costs related to failed exchanges

These are any additional average annual costs related to failed data exchanges that may be company-specific or not mentioned in this section.

Line 37 – Costs due to cancelled and/or rescheduled reviews

These are the average annual costs associated with time wasted by reviews that are failures due to corrupted or incomplete data. These costs include direct meeting costs such as attendees travelling, attending, and returning without being able to resolve issues because data are not available.

Line 38 – Cost due to delays in program progress caused by incomplete design reviews

These are the average annual costs that result from program delays caused by missing or inaccurate data. These average annual costs can be directly attributed to the use of missing or inaccurate data at design reviews. These costs could include program delay, wasted efforts, tooling expense, or other costs.

F.1.5 Costs of engineering changes[2]

The calculations in this section are primarily based on monthly numbers.

Line 40 – Number of Engineering Change Notices

This is the average monthly number of Engineering Change Notices (ECNs) that have been generated as a direct result of CAD data quality problems.

[2] The name for a formal engineering change varies from company to company. The term “ECN “ is used here to refer to all formal engineering changes.

Line 41 – Time spent implementing an ECN

This is the average cost associated with executing an ECN. This may involve issuing, processing, and submitting an ECN.

Line 42 – Time spent tracking the progress of an ECN

This is the average time spent per ECN tracking the progress of an ECN from start to finish.

Line 43 – Time spent re-sending an ECN

This is the average time spent re-processing or re-sending an ECN due to failures in the distribution process.

Line 44 – Time spent communicating ECN to the trading partners

This is the average time per ECN spent notifying affected trading partners of the ECN.

Line 49 – Other costs related to engineering changes

This is the estimated average annual additional costs related to engineering changes that might be company-specific or not already mentioned in this section.

F.1.6 Potential cost savings

The calculations in this section are primarily based on annual numbers.

Line 52 – Loss of business

This is the average annual cost of business lost due to CAD data quality–either current business or future business. This may be difficult to quantify. The idea is that if significant business has been lost due to problems with CAD data quality, that will probably be known and should be added to the overall cost of data quality problems.

Line 53 – Other costs

These are any additional average annual costs related to CAD data quality that might be company-specific or not mentioned in this template.

To calculate costs for addressing product data quality problems related to CAD-to-CAD data exchange, see the template in section F.1.45.

F.1.7 CAD-Related Costs of Product Data Quality

	1	Hourly labour cost (employee's rate + overhead)	1	$		
Costs of corrupted data prior to exchange or translation **(calculated on a monthly basis)**	2	Average number of CAD files or models worked on per month	2			
	3	Average time spent checking for corrupt data per file or model	3	hrs		
	4	Multiply line 3 by line Table 1. This is the monthly time spent checking data.	4	hrs		
	5	Number of corrupted files or models (prior to exchange) per month	5			
	6	Average time spent fixing or modifying data per file or model	6	hrs		
	7	Average time spent on data re-entry or restarting per file or model	7	hrs		
	8	Add lines 6 and 7. This is your total time spent per corrupted file or model.	8	hrs		
	9	Multiply line 8 by line 5. This is your total monthly time spent addressing corrupted data	9	hrs		
	10	Add lines 4 and 9. This is your average monthly labour addressing CAD data file corruption.	10	hrs		
	11	Multiply line 10 by line 1. This is your cost per month from corrupted data.	11	$		

	12	Multiply line 11 by 12 months. This is your yearly subtotal on corrupted data.	12	$			
	13	Annual costs due to errors resulting from data re-entry, modification, or re-creation	13	$			
	14	Other annual costs related to corrupt data	14	$			
	15	Add lines 12 through 14. This is your total cost per year on corrupted data.				15	$
Costs of failed CAD-to-CAD data exchanges **(calculated on a monthly basis)**	16	Monthly number of data exchanges (internal & external) per month	16				
	17	Monthly number of failed exchanges due to corrupted data	17				
	18	Monthly number of failed exchanges due to lack of data	18				
	19	Monthly number of failed exchanges due to too much data	19				
	20	Add lines 17 through 19. This is your total number of bad exchanges.	20				
	21	Average labour time spent checking a file or model for corrupt data	21	hrs			
	22	Average labour time spent checking a file or model for insufficient data	22	hrs			
	23	Average labour time spent checking a file or model for excessive data	23	hrs			
	24	Add lines 21 through 23. This is your total time spent checking each file or model.	24	hrs			
	25	Multiply line 24 by line 16. This is your total monthly time spent checking files or models.	25	hrs			
	26	Average time spent fixing or modifying data per exchange	26	hrs			
	27	Average time spent on data re-entry or restarting per exchange	27	hrs			
	28	Average time spent reprocessing, re-sending or receiving data per exchange	28	hrs			
	29	Add lines 26 through 28. This is your total time spent per failed exchange.	29	hrs			
	30	Multiply line 29 by line 20. This is your total time spent on resolving failed exchanges.	30	hrs			
	31	Add lines 25 & 30. This is your total time spent on bad exchanges.	31	hrs			
	32	Multiply line 31 by line 1. This is your cost per month from failed exchanges.	32	$			
	33	Multiply line 32 by 12 months. This is your yearly subtotal on failed exchanges.	33	$			
	34	Additional annual costs due to errors during data re-entry, modification, or re-creation.	34	$			
	35	Other annual costs related to failed exchanges.	35	$			
	36	Add lines 33 through 35. This is your total cost per year on failed exchanges.				36	$

Costs related to reviews	**37**	Annual costs due to cancelled and/or rescheduled reviews due to corrupted or incomplete data	**37**	**$**		
	38	Annual cost due to delays in program progress caused by missing or inaccurate data used in design reviews	**38**	**$**		
	39	Add lines 37 through 38. These are your annual costs related to design review problems.			**39**	**$**
Costs of engineer-ing changes	**40**	Annual monthly number of ECNs from CAD data quality problems	**40**			
	41	Average time spent implementing an ECN	**41**	**hrs**		
	42	Average time spent tracking an ECN	**42**	**hrs**		
	43	Average time spent re-sending versions of an ECN	**43**	**hrs**		
	44	Average time spent communicating an ECN to trading partner(s)	**44**	**hrs**		
	45	Add lines 41 through 44. This is your total time spent per ECN.	**45**	**hrs**		
	46	Multiply line 45 by line 1. This is your time cost per ECN.	**46**	**$**		
	47	Multiply line 46 by line 40. This is your monthly subtotal on ECNs.	**47**	**$**		
	48	Multiply line 47 by 12. This is your annual subtotal on ECNs.	**48**	**$**		
	49	Other annual costs related to engineering changes	**49**	**$**		
	50	Add lines 48 through 49. This is your total cost per year on ECNs.			**50**	**$**
Potential cost savings	**51**	Add lines 15, 36, 39, and 50. This is a subtotal of your total annual costs due to CAD-to-CAD exchange-related data quality problems.	**51**	**$**		
	52	Annual loss of business due to data quality issues	**52**	**$**		
	53	Other annual costs related to data quality	**53**	**$**		
	54	Add lines 51 through 53. These are your total potential annual cost savings.			**54**	**$**

F.1.8 Annual CAM-Related Costs of Product Data Quality

The following are the line-by-line instructions for completing the template F.1.15. for the costs of product data quality problems between CAD systems and CAM systems. These costs explicitly do not include costs that occur in other activities. Those other costs are addressed in the other templates. The line-by-line instructions describe what data go in the various places in the template. Calculated entries are described directly on the template rather than here.

F.1.9 Supporting information

Line 1 – Cost of labour

This is the cost rate per hour per employee. This includes employee's pay rate, benefits, equipment costs and overhead. This value is used in some of the later computations.

Line 2 – Number of components processed

This is the average annual number of distinct component designs processed by the CAM software. This is based on total effort per component design, including repeated work on the same primary design, regardless of the cause of the repeat.

F.1.10 Costs of failed CAD-to-CAM data exchanges

Most of the following costs are calculated on a per-component basis.

Line 3 – Number of data exchanges per component (internal and external)

This is the average number of CAD files or models that are sent to the CAM software for a component. Exchanges can be internal to the company; between employees in a department, or between departments, facilities, or divisions. External exchanges take place between trading partners.

Line 4 – Number of exchanges with corrupted data

This is the average number of CAD files or models that have failed to complete the exchange process for a typical component. The files or models are usually deemed unusable.

Line 5 – Number of exchanges with lack of data

This is the average number of CAD files or models that are found to be incomplete for a given component. The files or models are usually salvageable with modification or data re-entry.

Line 6 – Number of exchanges with too much data

This is the average number of CAD files or models that are found to contain too much detail for a given component. The files or models are usually salvageable with modification or data re-entry.

Line 8 – Checking a file or model for corrupt data

This is the average amount of labour time that is spent checking an exchanged file or model for file contamination. This normally will dictate whether the file or model is usable or salvageable.

Line 9 – Average time spent checking a file or model for insufficient data

This is the average amount of labour time that is spent checking an exchanged file or model for completeness of the data. Missing data normally need to be added to the CAD file or model.

Line 10 – Average time spent checking a file or model for excessive data

This is the average amount of labour time that is spent checking an exchanged file or model for data that are extraneous to the purpose for which the data are to be used. Excess data often need to be stripped out of the CAD file or model.

Line 13 – Average time spent fixing or modifying data per exchange

This is the average amount of time that is spent trying to fix files or models for which the exchange has failed, either by fixing or removing the excessive or corrupt data. The file or model may be salvageable after it has been modified.

Line 14 – Average time spent on data re-entry or restarting per exchange

This is the average amount of time that is spent re-entering the corrupted or missing geometry. If the exchanged file or model is determined to be corrupted, unusable, or unsalvageable, creating a new file or model may be necessary.

Line 15 – Average time spent reprocessing, re-sending or receiving data per exchange

This is the average amount of time that is spent to reiterate the exchange process (not including the original try at sending the data). If the exchanged file or model is determined to be unusable or unsalvageable, repeating the exchange may be necessary. This includes any prep time, processing time for translations, and time spent receiving or re-sending the exchanged CAD data.

Line 21 – Costs due to data re-entry, modification, or re-creation errors

These are the average annual costs related to errors or mistakes made during the data re-entry or re-creation process. If data are entered incorrectly, this may result in additional expense. One way to calculate this value is to estimate the error rate of corrected or recreated data, then multiply that by the Line 20 value.

Line 22 – Other costs related to failed exchanges

These are any additional average annual costs related to failed data exchanges that may be company-specific or not mentioned in this section.

F.1.11 Costs related to manufacturing issues

The calculations in this section are primarily based on per-component numbers.

Line 24 – Corrective tooling

This is the average per-component cost related to additional tooling or changes that may be required resulting from CAD data quality problems. These costs occur only if tooling has been started or completed before corrective action must be taken.

Line 25 – Scrap costs

This is the average per-component cost related to parts being rejected or discarded based on the use of poor-quality CAD data.

Line 26 – Excess prototyping costs

These are the average per-component costs that result from having to redo digital mock-up work or make more digital mock-ups due to CAD data quality problems.

Line 29 – Other costs related to manufacturing issues

These are any additional average annual costs related to manufacturing that might be company-specific or not mentioned in this section.

F.1.12 Costs related to scheduling and delivery issues

The calculations in this section are primarily based on per-component numbers.

Line 31 – Excessive overtime costs

This is the average per-component cost related to employees working additional hours to meet scheduling or delivery demands.

Line 32 – Missed delivery date costs

This is the average per-component cost related to any penalties or fines for missed deliveries or delays in the schedule.

Line 33 – Outsourcing costs

This is the average per-component cost related to outsourcing work or services to meet scheduling or delivery demands.

Line 36 – Costs due to cancelled and/or rescheduled reviews

These are the average annual costs associated with time wasted by reviews that are failures due to missing CAM data that, in turn, resulted from corrupted or incomplete data. These costs include direct meeting costs such as attendees travelling, attending, and returning without being able to resolve issues because data are not available.

Line 37 – Costs due to delays in program progress caused by incomplete design reviews

These are the average annual costs that result from program delays caused by CAM problems due to missing or inaccurate data. These average annual costs can be indirectly attributed to the use of missing or inaccurate data at design reviews. These costs could include program delay, wasted efforts, tooling expense, or other costs.

Line 38 – Other costs related to scheduling and delivery issues

These are any additional average annual costs related to scheduling and delivery that might be company-specific or not mentioned in this section.

F.1.13 Costs of engineering changes

The calculations in this section are primarily based on per-component numbers.

Line 40 – Number of Engineering Change Notices

This is the average per-component number of Engineering Change Notices (ECNs) that have been generated as a direct result of CAD data quality problems.

Line 41 – Time spent implementing an ECN

This is the average cost associated with executing an ECN. This may involve issuing, processing, and submitting an ECN.

Line 42 – Time spent tracking the progress of an ECN

This is the average time spent per ECN tracking the progress of an ECN from start to finish.

Line 43 – Time spent re-sending an ECN

This is the average time spent re-processing or re-sending an ECN due to failures in the distribution process.

Line 44 – Time spent communicating an ECN to trading partner(s)

This is the average time per ECN spent notifying affected trading partners of the ECN.

Line 49 – Other costs related to engineering changes

This is the estimated average annual additional cost related to engineering changes that might be company-specific or not already mentioned in this section.

F.1.14 Potential cost savings

The calculations in this section are primarily based on annual numbers.

Line 52 – Loss of business

This is the average annual cost of business lost due to CAD data quality, either current business or future business.

Line 53 – Other costs

These are any additional average annual costs related to CAD data quality that might be company-specific or not mentioned in this template.

To calculate costs for addressing product data quality problems related to CAD-to-CAM data exchange, see template F.1.15.

F.1.15 CAM-Related Costs of Product Data Quality

	1	Hourly labour cost (employee's rate + overhead)	**1**	$		
	2	Average annual number of components processed	**2**			
Costs of failed CAD-to-CAM data exchanges (calculated on a per component basis)	**3**	Average number of CAD-to-CAM data exchanges (internal and external) per component	**3**			
	4	Number of failed exchanges due to corrupted data per component	**4**			
	5	Number of failed exchanges due to lack of data per component	**5**			
	6	Number of failed exchanges due to too much data per component	**6**			
	7	Add lines 4 through 6. This is your total number of bad exchanges per component.	**7**			
	8	Average time spent checking a file or model for corrupt data	**8**	**hrs**		
	9	Average time spent checking a file or model for insufficient data	**9**	**hrs**		
	10	Average time spent checking a file or model for excessive data	**10**	**hrs**		
	11	Add lines 8 through 10. This is your total time per exchange spent checking for problems.	**11**	**hrs**		
	12	Multiply line 11 by line 3. This is your time per component spent on checking for problems.	**12**	**hrs**		
	13	Average time spent fixing or modifying data per exchange	**13**	**hrs**		
	14	Average time spent on data re-entry or restarting per exchange	**14**	**hrs**		
	15	Average time spent reprocessing, re-sending or receiving data per exchange	**15**	**hrs**		
	16	Add lines 13 through 15. This is your total time spent fixing each failed exchange.	**16**	**hrs**		
	17	Multiply line 16 by line 7. This is your total time spent on resolving failed exchanges.	**17**	**hrs**		
	18	Add lines 12 & 17. This is your total time per component spent addressing exchange problems.	**18**	**hrs**		
	19	Multiply line 18 by line 2. This is your annual time spent on failed exchanges.	**19**	**hrs**		
	20	Multiply line 19 by line 1. This is your annual subtotal on failed exchanges.	**20**	$		
	21	Annual costs due to additional errors during data re-entry, modification or re-creation	**21**	$		
	22	Other annual costs related to failed CAD-to-CAM exchanges	**22**	$		
	23	Add lines 20 through 23. This is your total cost per year on failed exchanges.			**23**	**$**

Costs related to manufac-turing issues	**24**	Average per-component corrective tooling costs due to data quality problems	**24**	**$**		
	25	Average per-component scrap costs due to data quality problems	**25**	**$**		
	26	Average per-component excess prototyping costs due to data quality problems	**26**	**$**		
	27	Add line 24 through line 26. This is the per-component cost from manufacturing issues.	**27**	**$**		
	28	Multiply line 27 by line 2. This is the annual cost from these sources.	**28**	**$**		
	29	Other annual costs related to manufacturing issues	**29**	**$**		
	30	Add lines 28 and 29. These are your total costs per year on manufacturing issues.			**30**	**$**
Costs related to scheduling and delivery issues	**31**	Average per-component excessive overtime costs	**31**	**$**		
	32	Average per-component missed delivery date costs	**32**	**$**		
	33	Average per-component outsourcing costs	**33**	**$**		
	34	Add line 31 through line 33. This is the per-component cost from scheduling and delivery issues.	**34**	**$**		
	35	Multiply line 34 by line 2. This is the annual cost from scheduling and delivery issues.	**35**	**$**		
	36	Annual costs due to cancelled and/or rescheduled reviews, due to corrupted or incomplete data	**36**	**$**		
	37	Annual cost due to delays in program progress caused by missing or inaccurate data used in design reviews	**37**	**$**		
	38	Other annual costs related to scheduling and delivery issues	**38**	**$**		
	39	Add lines 35 through 38. These are your total costs related to scheduling and delivery issues.			**39**	**$**
Costs of engineer-ing changes	**40**	Average per component number of ECNs from data quality problems	**40**			
	41	Average time spent implementing an ECN	**41**	**hrs**		
	42	Average time spent tracking an ECN	**42**	**hrs**		
	43	Average time spent re-sending versions of an ECN	**43**	**hrs**		
	44	Average time spent communicating an ECN to trading partner(s)	**44**	**hrs**		
	45	Add lines 41 through 44. This is your total time spent per ECN.	**45**	**hrs**		
	46	Multiply line 45 by line 1. This is your time cost per ECN.	**46**	**$**		
	47	Multiply line 46 by line 40. This is your per-component subtotal on ECNs.	**47**	**$**		
	48	Multiply line 47 by line 2. This is your yearly subtotal on ECNs.	**48**	**$**		
	49	Other annual costs related to engineering changes	**49**	**$**		
	50	Add lines 48 and 49. This is your total cost per year on ECNs due to data problems.			**50**	**$**
Potential cost savings	**51**	Add lines 23, 30, 39, and 50. This is a subtotal of your total annual costs.	**51**	**$**		
	52	Annual loss of business due to data quality issues	**52**	**$**		
	53	Other annual costs related to data quality	**53**	**$**		
	54	Add lines 51 through 53. These are your total potential annual cost savings.			**54**	**$**

F.1.16 Annual CAE-Related Costs of Product Data Quality

The following are the line-by-line instructions for completing the template F.1.21. for the costs of product data quality problems between CAD systems and CAE systems. These costs explicitly do not include costs that occur in other activities. Those other costs are addressed in the other templates. The line-by-line instructions describe what data go in the various places in the template. Calculated entries are described directly on the template rather than here.

F.1.17 Supporting information

Line 1 – Cost of labour

This is the cost rate per hour per employee. This includes employee's pay rate, benefits, equipment costs, and overhead. This value is used in some of the later computations.

Line 2 – Number of analyses processed

This is the average annual number of analyses processed by the CAE software. The following calculations are largely based on effort per analysis.

F.1.18 Costs of failed CAD-to-CAE data exchanges

Line 3 – Number of data exchanges per analysis (internal and external)

This is the average number of CAD files or models that are sent to the CAE software for an analysis. Exchanges can be internal to the company; between employees in a department; or between departments, facilities, or divisions. External exchanges take place between trading partners.

Line 4 – Number of exchanges with corrupted data

This is the average number of CAD files or models that have failed to complete the exchange process for a typical analysis. The files or models are usually deemed unusable.

Line 5 – Number of exchanges with lack of data

This is the average number of CAD files or models that are found to be incomplete for a given analysis. The files or models are usually salvageable with modification or data re-entry.

Line 6 – Number of exchanges with too much data

This is the average number of CAD files or models that are found to contain too much detail for a given analysis. The files or models are usually salvageable with modification or data re-entry.

Line 8 – Checking a file or model for corrupt data

This is the average amount of labour time that is spent checking an exchanged file or model for file contamination. This normally will dictate whether the file or model is usable or salvageable.

Line 9 – Average time spent checking a file or model for insufficient data

This is the average amount of labour time that is spent checking an exchanged file or model for completeness of the data. Missing data normally need to be added to the CAD file or model.

Line 10 – Average time spent checking a file or model for excessive data

This is the average amount of labour time that is spent checking an exchanged file or model for data that are extraneous to the purpose for which the data are to be used. Excess data often need to be stripped out of the CAD file or model.

Line 13 – Average time spent fixing or modifying data per exchange

This is the average amount of time that is spent trying to fix files or models for which the exchange has failed, either by fixing or removing the excessive or corrupt data. The file or model may be salvageable after it has been modified.

Line 14 – Average time spent on data re-entry or restarting per exchange

This is the average amount of time that is spent re-entering the corrupted or missing geometry. If the exchanged file or model is determined to be corrupted, unusable or unsalvageable, creating a new file or model may be necessary.

Line 15 – Average time spent reprocessing, re-sending, or receiving data per exchange

This is the average amount of time that is spent to reiterate the exchange process (not including the original try at sending the data). If the exchanged file or model is determined to be unusable or unsalvageable, repeating the exchange may be necessary. This includes any prep time, processing time for translations, and time spent receiving or re-sending the exchanged CAD data.

Line 21 – Costs due to data re-entry, modification, or re-creation errors

These are the average annual costs related to errors or mistakes made during the data re-entry or re-creation process. If data are entered incorrectly, this may result in additional expenses. One way to calculate this value is to estimate the error rate of corrected or recreated data, then multiply that by the Line 20 value.

Line 22 – Other costs related to failed exchanges

These are any additional average annual costs related to failed data exchanges that may be company-specific or not mentioned in this section.

F.1.19 Costs related to scheduling and delivery issues

Line 24 – Excessive overtime costs

This is the average per-analysis cost related to employees working additional hours to meet scheduling or delivery demands.

Line 25 – Missed delivery date costs

This is the average per-analysis cost related to any penalties or fines for missed deliveries or delays in the schedule.

Line 26 – Outsourcing costs

This is the average per-analysis cost related to outsourcing work or services to meet scheduling or delivery demands.

Line 29 – Costs due to cancelled and/or rescheduled reviews

These are the average annual costs associated with time wasted by reviews that are failures due to missing CAE analysis results that, in turn, resulted from corrupted or incomplete data. These costs include direct meeting costs such as attendees travelling, attending, and returning without being able to resolve issues because data are not available.

Line 30 – Cost due to delays in program progress caused by incomplete design reviews

These are the average annual costs that result from program delays caused by CAE analysis problems due to missing or inaccurate data. These average annual costs can be indirectly attributed to the use of missing or inaccurate data at design reviews. These costs could include program delay, wasted efforts, tooling expense, or other costs.

Line 31 – Other costs related to scheduling and delivery issues

These are any additional average annual costs related to scheduling and delivery that might be company-specific or not mentioned in this section.

F.1.20 Potential cost savings

Line 34 – Loss of business

This is the average annual cost of business lost due to CAD data quality, either current business or future business.

Line 35 – Other costs

These are any additional average annual costs related to CAD data quality that might be company-specific or not mentioned in this template.

To calculate costs for addressing product data quality problems related to CAD-to-CAE data exchange, see template F.1.21 .

F.1.21 Annual CAE-related Costs of Product Data Quality

	1	Hourly labour cost (employee's rate + overhead)	1	$		
	2	Average annual number of CAE analyses processed	2			
Costs of failed CAD-to-CAE data exchanges (calculated on a per analysis basis)	3	Average number of CAD to CAE data exchanges (internal and external) per analysis	3			
	4	Number of failed exchanges due to corrupted data per analysis	4			
	5	Number of failed exchanges due to lack of data per analysis	5			
	6	Number of failed exchanges due to too much data per analysis	6			
	7	Add lines 4 through 6. This is your total number of bad exchanges per analysis.	7			
	8	Average time spent checking a file or model for corrupt data	8	hrs		
	9	Average time spent checking a file or model for insufficient data	9	hrs		
	10	Average time spent checking a file or model for excessive data	10	hrs		
	11	Add lines 8 through 10. This is your total time per exchange spent checking for problems.	11	hrs		
	12	Multiply line 11 by line 3. This is your time per analysis spent on checking for problems.	12	hrs		
	13	Average time spent fixing or modifying data per exchange	13	hrs		
	14	Average time spent on data re-entry or restarting per exchange	14	hrs		
	15	Average time spent reprocessing, re-sending, or receiving data per exchange	15	hrs		
	16	Add lines 13 through 15. This is your total time spent fixing each failed exchange.	16	hrs		
	17	Multiply line 16 by line 7. This is your total time spent resolving failed exchanges.	17	hrs		
	18	Add lines 12 and 17. This is your total time per analysis spent addressing exchange problems.	18	hrs		
	19	Multiply line 18 by line 2. This is your annual time spent on failed exchanges.	19	hrs		
	20	Multiply line 19 by line 1. This is your annual subtotal on failed exchanges.	20	$		
	21	Annual costs due to additional errors during data re-entry, modification, or re-creation.	21	$		

	22	Other annual costs related to failed CAD-to-CAE exchanges	22	$		
	23	Add lines 20 through 22. This is your total cost per year on failed exchanges.			23	$
Costs related to analysis scheduling and delivery issues	24	Average per-analysis excessive overtime costs	24	$		
	25	Average per-analysis missed delivery date costs	25	$		
	26	Average per analysis outsourcing costs	26	$		
	27	Add lines 24 through 26. This is the per-analysis cost from scheduling and delivery issues	27	$		
	28	Multiply line 27 by line 2. This is the annual cost from scheduling and delivery issues.	28	$		
	29	Annual costs due to cancelled and/or rescheduled reviews due to corrupted or incomplete data	29	$		
	30	Annual cost due to delays in program progress caused by missing or inaccurate data used in design reviews	30	$		
	31	Other annual costs related to scheduling and delivery issues	31	$		
	32	Add lines 28 through 31. These are your total annual costs related to scheduling and delivery issues.			32	$
Potential cost savings	33	Add lines 23 and 32. This is a subtotal of your total annual costs	33	$		
	34	Annual loss of business due to data quality issues	34	$		
	35	Other annual costs related to data quality	35	$		
	36	Add lines 33 through 35. These are your total potential annual cost savings.			36	$

F.1.22 Annual Prototype-Related Costs of Product Data Quality

The following are line-by-line instructions for completing the template F.1.27. for the costs of product data quality problems between CAD systems and computer-driven prototyping systems. These costs explicitly do not include costs that occur in other activities. Those other costs are addressed in the other templates. The line-by-line instructions describe what data go in the various places in the template. Calculated entries are described directly on the template rather than here.

F.1.23 Supporting information

Line 1 – Cost of labour

This is the cost rate per hour per employee. This includes employee's pay rate, benefits, equipment costs, and overhead. This value is used in some of the later computations.

Line 2 – Number of prototypes built

This is the average annual number of prototypes built. The following calculations are largely based on effort per prototype.

F.1.24 Costs of failed CAD-to-Prototyping data exchanges

Line 3 – Number of data exchanges per prototype (internal and external)

This is the average number of CAD files or models that are sent to the prototyping software for a prototype. Exchanges can be internal to the company, between employees in a department; or between departments, facilities, or divisions. External exchanges take place between trading partners.

Line 4 – Number of exchanges with corrupted data

This is the average number of CAD files or models that have failed to complete the exchange process for a typical prototype. The files or models are usually deemed unusable.

Line 5 – Number of exchanges with lack of data

This is the average number of CAD files or models that are found to be incomplete for a given prototype. The files or models are usually salvageable with modification or data re-entry.

Line 6 – Number of exchanges with too much data

This is the average number of CAD files or models that are found to contain too much detail for a given prototype. The files or models are usually salvageable with modification or data re-entry.

Line 8 – Checking a file or model for corrupt data

This is the average amount of labour time that is spent checking an exchanged file or model for file contamination. This normally will dictate whether the file or model is usable or salvageable.

Line 9 – Average time spent checking a file or model for insufficient data

This is the average amount of labour time that is spent checking an exchanged file or model for completeness of the data. Missing data normally need to be added to the CAD file or model.

Line 10 – Average time spent checking a file or model for excessive data

This is the average amount of labour time that is spent checking an exchanged file or model for data that is extraneous to the purpose for which the data are to be used. Excess data often need to be stripped out of the CAD file or model.

Line 13 – Average time spent fixing or modifying data per exchange

This is the average amount of time that is spent trying to fix files or models for which the exchange has failed, either by fixing or removing the excessive or corrupt data. The file or model may be salvageable after it has been modified.

Line 14 – Average time spent on data re-entry or restarting per exchange

This is the average amount of time that is spent re-entering the corrupted or missing geometry. If the exchanged file or model is determined to be corrupted, unusable, or unsalvageable, creating a new file or model may be necessary.

Line 15 – Average time spent reprocessing, re-sending or receiving data per exchange

This is the average amount of time that is spent to reiterate the exchange process not including the original attempt to send the data). If the exchanged file or model is determined to be unusable or unsalvageable, repeating the exchange may be necessary. This includes any prep time, processing time for translations, and time spent receiving or re-sending the exchanged CAD data.

Line 21 – Costs due to data re-entry, modification, or re-creation errors

These are the average annual costs related to errors or mistakes made during the data re-entry or re-creation process. If data are entered incorrectly, this may result in additional expenses. One way to calculate this value is to estimate the error rate of corrected or recreated data, then multiply that by the Line 20 value.

Line 22 – Other costs related to failed exchanges

These are any additional average annual costs related to failed data exchanges that may be company-specific or not mentioned in this section.

F.1.25 Costs related to scheduling and delivery issues

Line 24 – Excessive overtime costs

This is the average per-prototype cost related to employees working additional hours to meet scheduling or delivery demands.

Line 25 – Missed delivery date costs

This is the average per-prototype cost related to any penalties or fines for missed deliveries or delays in the schedule.

Line 26 – Outsourcing costs

This is the average per-prototype cost related to outsourcing work or services to meet scheduling or delivery demands.

Line 29 – Costs due to cancelled and/or rescheduled reviews

These are the average annual costs associated with time wasted by reviews that are failures due to missing prototype results that, in turn, resulted from corrupted or incomplete data. These costs include direct meeting costs such as attendees travelling, attending, and returning without being able to resolve issues because data are not available.

Line 30 – Cost due to delays in program progress caused by incomplete design reviews

These are the average annual costs that result from program delays caused by prototype problems due to missing or inaccurate data. These average annual costs can be indirectly attributed to the use of missing or inaccurate data at design reviews. These costs could include program delay, wasted efforts, tooling expense, or other costs.

Line 31 – Other costs related to scheduling and delivery issues

These are any additional average annual costs related to scheduling and delivery that might be company-specific or not mentioned in this section.

F.1.26 Potential cost savings

Line 34 – Loss of business

This is the average annual cost of business lost due to CAD data quality, either current business or future business.

Line 35 – Other costs

These are any additional average annual costs related to CAD data quality that might be company-specific or not mentioned in this template.

To calculate costs for addressing product data quality problems related to CAD-to-prototype data exchange, see template F.1.27.

F.1.27 Annual Prototyping Costs of Product Data Quality

	1	Hourly labour cost (employee's rate + overhead)	1	$		
	2	Average annual number of prototypes made	2			
Costs of failed CAD-to-Prototyping data exchanges **(calculated on a per prototype basis)**	3	Average number of CAD-to-prototyping data exchanges (internal and external) per prototype	3			
	4	Number of failed exchanges due to corrupted data per prototype	4			
	5	Number of failed exchanges due to lack of data per prototype	5			
	6	Number of failed exchanges due to too much data per prototype	6			
	7	Add lines 4 through 6. This is your total number of bad exchanges per prototype.	7			
	8	Average time spent checking a file or model for corrupt data	8	hrs		
	9	Average time spent checking a file or model for insufficient data	9	hrs		
	10	Average time spent checking a file or model for excessive data	10	hrs		
	11	Add lines 8 through 10. This is your total time per exchange spent checking for problems.	11	hrs		
	12	Multiply line 11 by line 3. This is your time per prototype spent on checking for problems.	12	hrs		
	13	Average time spent fixing or modifying data per exchange	13	hrs		
	14	Average time spent on data re-entry or restarting per exchange	14	hrs		
	15	Average time spent reprocessing, re-sending or receiving data per exchange	15	hrs		
	16	Add lines 13 through 15. This is your total time spent fixing each failed exchange.	16	hrs		
	17	Multiply line 16 by line 7. This is your total time spent resolving failed exchanges.	17	hrs		
	18	Add lines 12 and 17. This is your total time per prototype spent addressing exchange problems.	18	hrs		
	19	Multiply line 18 by line 2. This is your annual time spent on failed exchanges	19	hrs		
	20	Multiply line 19 by line 1. This is your annual subtotal on failed exchanges.	20	$		
	21	Annual costs due to additional errors during data re-entry, modification or re-creation	21	$		
	22	Other annual costs related to failed CAD-to-prototyping exchanges	22	$		
	23	Add lines 20 through 22. This is your total cost per year on failed exchanges.			23	$
Costs related to prototype scheduling and delivery issues	24	Average per-prototype excessive overtime costs	24	$		
	25	Average per-prototype missed delivery date costs	25	$		
	26	Average per-prototype outsourcing costs	26	$		
	27	Add lines 24 through 26. This is the per prototype cost from scheduling and delivery issues	27	$		
	28	Multiply line 27 by line 2. This is the annual cost from scheduling and delivery issues.	28	$		
	29	Annual costs due to cancelled and/or rescheduled reviews due to corrupted or incomplete data	29	$		
	30	Annual cost due to delays in program progress caused by missing or inaccurate data used in design reviews	30	$		
	31	Other annual costs related to scheduling and delivery issues	31	$		
	32	Add lines 28 through 31. These are your total annual costs related to scheduling and delivery issues.			32	$
Potential cost savings	33	Add lines 23 and 32. This is a subtotal of your total annual costs.	33	$		
	34	Annual loss of business due to data quality issues	34	$		
	35	Other annual costs related to data quality	35	$		
	36	Add lines 33 through 35. These are your total potential annual cost savings.			36	$

F.1.28 Annual Digital-Mock-up-Related Costs of Product Data Quality

The following are the line-by-line instructions for completing the template F.1.33. for the costs of product data quality problems between CAD systems and digital mock-up (virtual assembly) systems. These costs explicitly do not include costs that occur in other activities. Those other costs are addressed in the other templates. The line-by-line instructions describe what data go in the various places in the template. Calculated entries are described directly on the template rather than here.

F.1.29 Supporting information

Line 1 – Cost of labour

This is the cost rate per hour per employee. This includes employee's pay rate, benefits, equipment costs, and overhead. This value is used in some of the later computations.

F.1.30 Costs of failed CAD-to-digital-mock-up data exchanges

Line 2 – Number of digital mock-ups processed per month

This is the average monthly number of digital mock-ups created. The following calculations are largely based on total effort per digital mock-up.

Line 3 – Number of data exchanges per digital mock-up (internal and external)

This is the average number of CAD files or models that are sent to the digital mock-up software for a digital mock-up. Exchanges can be internal to the company between employees in a department or between departments, facilities, or divisions. External exchanges take place between trading partners.

Line 4 – Number of exchanges with corrupted data

This is the average number of CAD files or models that have failed to complete the exchange process for a typical digital mock-up. The files or models are usually deemed unusable.

Line 5 – Number of exchanges with lack of data

This is the average number of CAD files or models that are found to be incomplete for a typical digital mock-up. The files or models are usually salvageable with modification or data re-entry.

Line 6 – Number of exchanges with too much data

This is the average number of CAD files or models that are found to contain too much detail for a typical digital mock-up. The files or models are usually salvageable with modification or data re-entry.

Line 8 – Checking a file or model for corrupt data

This is the average amount of labour time that is spent checking an exchanged file or model for file contamination. This normally will dictate whether the file or model is usable or salvageable.

Line 9 – Average time spent checking a file or model for insufficient data

This is the average amount of labour time that is spent checking an exchanged file or model for completeness of the data. Missing data normally need to be added to the CAD file or model.

Line 10 – Average time spent checking a file or model for excessive data

This is the average amount of labour time that is spent checking an exchanged file or model for data that is extraneous to the purpose for which the data are to be used. Excess data often need to be stripped out of the CAD file or model.

Line 13 – Average time spent fixing or modifying data per exchange

This is the average amount of time that is spent trying to fix files or models, for which the exchange has failed, either by fixing or removing the excessive or corrupt data. The file or model may be salvageable after it has been modified.

Line 14 – Average time spent on data re-entry or restarting per exchange

This is the average amount of time that is spent re-entering the corrupted or missing geometry. If the exchanged file or model is determined to be corrupted, unusable, or unsalvageable, creating a new file or model may be necessary.

Line 15 – Average time spent reprocessing, re-sending, or receiving data per exchange

This is the average amount of time that is spent to reiterate the exchange process (not including the original attempt to send the data). If the exchanged file or model is determined to be unusable or unsalvageable, repeating the exchange may be necessary. This includes any prep time, processing time for translations, and time spent receiving or re-sending the exchanged CAD data.

Line 21 – Costs due to data re-entry, modification, or re-creation errors

These are the average annual costs related to errors or mistakes made during the data re-entry or re-creation process. If data are entered incorrectly, this may result in additional expenses. One way to calculate this value is to estimate the error rate of corrected or recreated data, and then multiply that by the Line 20 value.

Line 22 – Other costs related to failed exchanges

These are any additional average annual costs related to failed data exchanges that may be company-specific or not mentioned in this section.

F.1.31 Costs related to scheduling and delivery issues

Line 24 – Excessive overtime costs

This is the average per-digital-mock-up cost related to employees working additional hours to meet scheduling or delivery demands.

Line 25 – Missed delivery date costs

This is the average cost per digital mock-up related to any penalties or fines for missed deliveries or delays in the schedule.

Line 26 – Outsourcing costs

This is the average cost per digital mock-up related to outsourcing work or services to meet scheduling or delivery demands.

Line 29 – Costs due to cancelled and/or rescheduled reviews

These are the average annual costs associated with time wasted by reviews that are failures due to corrupted or incomplete data. These costs include direct meeting costs such as attendees travelling, attending, and returning without being able to resolve issues because data are not available.

Line 30 – Cost due to delays in program progress caused by incomplete design reviews

These are the costs that result from program delays caused by missing or inaccurate data. These average annual costs can be directly attributed to the use of missing or inaccurate data at design reviews. These costs could include program delay, wasted efforts, tooling expense, or other costs.

Line 31 – Other costs related to scheduling and delivery issues

These are any additional average annual costs related to scheduling and delivery that might be company-specific or not mentioned in this section.

F.1.32 Potential cost savings

Line 34 – Loss of business

This is the average annual cost of business lost due to CAD data quality, either current business or future business.

Line 35 – Other costs

These are any additional average annual costs related to CAD data quality that might be company-specific or not mentioned in this template.

To calculate costs for addressing product data quality problems related to CAD-to-digital-mock-up data exchange, see template F.1.33.

F.1.33 Annual Digital Mock-up Costs of Product Data Quality

	1	Hourly labour cost (employee’s rate + overhead)	1	$		
Costs of failed CAD- to-Digital Mock-up exchanges **(calculated on a per- digital-mock-up basis)**	2	Average monthly number of digital mock-ups made	2			
	3	Average number of CAD-to-digital mock-up data exchanges (internal and external) per digital mock-up	3			
	4	Number of failed exchanges due to corrupted data per digital mock-up	4			
	5	Number of failed exchanges due to lack of data per digital mock-up	5			
	6	Number of failed exchanges due to too much data per digital mock-up	6			
	7	Add lines 4 through 6. This is your total number of bad exchanges per digital mock-up.	7			
	8	Average time spent checking a file or model for corrupt data	8	hrs		
	9	Average time spent checking a file or model for insufficient data	9	hrs		
	10	Average time spent checking a file or model for excessive data	10	hrs		
	11	Add lines 8 through 10. This is your total time per exchange spent checking for problems.	11	hrs		
	12	Multiply line 11 by line 3. This is your time per digital mock-up spent on checking for problems.	12	hrs		
	13	Average time spent fixing or modifying data per bad exchange	13	hrs		
	14	Average time spent on data re-entry or restarting per bad exchange	14	hrs		
	15	Average time spent reprocessing, re-sending, or receiving data per bad exchange	15	hrs		
	16	Add lines 13 through 15. This is your total time spent fixing each failed exchange.	16	hrs		
	17	Multiply line 16 by line 7. This is your total time spent on resolving failed exchanges.	17	hrs		
	18	Add lines 12 and 17. This is your total time per digital mock-up spent addressing exchange problems.	18	hrs		
	19	Multiply line 18 by 12. This is your annual time spent on failed exchanges.	19	hrs		

	20	Multiply line 19 by line 1. This is your annual subtotal on failed exchanges.	20	$		
	21	Annual costs due to additional errors during data re-entry, modification. or re-creation	21	$		
	22	Other annual costs related to failed CAD-to-digital mock-up exchanges	22	$		
	23	Add lines 20 through 22. This is your total cost per year on failed exchanges.			23	$
Costs related to digital mock-up scheduling and delivery issues	24	Average per-digital-mock-up excessive overtime costs	24	$		
	25	Average per-digital-mock-up missed delivery date costs	25	$		
	26	Average per-digital-mock-up outsourcing costs	26	$		
	27	Add lines 24 through 26. This is the per-digital-mock-up cost from scheduling and delivery issues.	27	$		
	28	Multiply line 27 by line 2. This is the annual cost from scheduling and delivery issues.	28	$		
	29	Annual costs due to cancelled and/or rescheduled reviews due to corrupted or incomplete data	29	$		
	30	Annual cost due to delays in program progress caused by missing or inaccurate data used in design reviews	30	$		
	31	Other annual costs related to scheduling and delivery issues	31	$		
	32	Add lines 28 through 31. These are your total annual costs related to scheduling and delivery issues.			32	$
Potential cost savings	33	Add lines 23 and 32. This is a subtotal of your total annual costs	33	$		
	34	Annual loss of business due to digital mock-up data quality issues	34	$		
	35	Other annual costs related to digital mock-up data quality	35	$		
	36	Add lines 33 through 35. These are your total potential annual cost savings from improving data quality for digital mock-up.			36	$

F.1.34 PDM-Related Product Data Quality Costs

The following are the line-by-line instructions for completing the template F.1.44. for the costs of product data quality problems related to PDM systems. These costs explicitly do not include costs that occur in other activities. Those other costs are addressed in the other templates. The line-by-line instructions describe what data go in the various places in the template. Calculated entries are described directly on the template rather than here.

F.1.35 Supporting information

Line 1 – Cost of labour

This is the cost rate per hour per employee. This includes employee's pay rate, benefits, equipment costs, and overhead. This value is used in some of the later computations.

F.1.36 Costs of failed PDM data exchanges

Line 2 – Number of PDM data exchanges per month (internal and external)

This is the average number of PDM files that are exchanged in a given month. A PDM exchange can take place either between similar PDM systems in native format or translated between dissimilar PDM systems. Exchanges can be internal to the company; within or between departments, facilities or divisions. External exchanges take place between trading partners. This value is used for comparison.

Line 3 – Number of exchanges with corrupt data values

This is the average number of PDM files that have failed to complete the exchange process in a given month due to corrupt data values, typically bad strings or numeric values. These values are the attributes that define a PDM data object. Incorrect associations between PDM data objects should *not* be included in this number as they are addressed later in this template.

Line 4 – Average time spent checking an imported PDM model for corrupt data values

This is the average amount of labour time per exchange that is spent checking an imported file or model for correctness of the data values. Missing data normally need to be added to the PDM model.

Line 5 – Average time spent acquiring correct data values

This is the average amount of labour time per exchange that is spent trying to acquire correct replacement values for the corrupt data values that led to the failed exchange.

Line 6 – Average time spent on data value re-entry

This is the average amount of labour time per exchange that is spent re-entering the correct replacement values for imported PDM models for which the exchange has failed due to corrupt data values.

Line 7 – Average time lost using corrupt data values

This is the average amount of time per exchange that is lost as a result of work done on the basis of imported PDM models that contain bad data values. This is work done prior to discovery of corrupt data values that must be re-done in light of the correct data values. It includes both design as well as manufacturing re-work.

Line 10 – Number of exchanges with corrupt association between part meta-data and ECN information

This is the average number of PDM files that have failed to complete the exchange process in a given month due to corrupt associations between part information meta-data and related Engineering Change Notification (ECN) meta-data. Incorrect data values related to part or ECN information should *not* be included in this number as they are accounted for earlier in this template.

Line 11 – Average time spent locating the relevant ECN information

This is the average amount of labour time per exchange that is spent trying to acquire the relevant ECN information for imported PDM models to replace corrupt association between part meta-data and ECN information.

Line 12 – Average time spent on ECN information re-entry

This is the average amount of labour time per exchange that is spent re-entering relevant ECN information or re-associating the relevant ECN information with the correct part meta-data.

Line 13 – Average time lost working with part information that is outdated (has already been changed)

This is the average amount of labour time per exchange that is lost as a result of work done on the basis of imported PDM models that contain bad or missing ECN information. This is work done prior to discovery of corrupt ECN information that must be re-done in light of the correct ECN information. It includes both design as well as manufacturing re-work.

Line 14 – Average time spent re-engineering work already done under a previous ECN

This is the average amount of labour time per exchange that is spent re-engineering a design that was in fact already done in response to a previous but lost ECN.

Line 17 – Number of exchanges with corrupt association between part meta-data and authorisation information

This is the average number of PDM files that have failed to complete the exchange process in a given month due to corrupt associations between part information meta-data and related authorisation meta-data. Incorrect data values related to part or authorisation information should *not* be included in this number as they are accounted for earlier in this template.

Line 18 – Average time spent acquiring the relevant authorisation information

This is the average amount of labour time per exchange that is spent trying to acquire and verify the relevant authorisation information for imported PDM models with corrupt association between part meta-data and authorisation information.

Line 19 – Average time spent on authorisation information re-entry

This is the average amount of labour time per exchange that is spent re-entering relevant authorisation information or re-associating the relevant authorisation information with the correct part meta-data.

Line 22 – Number of exchanges with corrupt association between part meta-data and external bulk data

This is the average number of PDM files that have failed to complete the exchange process in a given month due to corrupt associations between part information meta-data and related meta-data describing external bulk data, typically digital files. Incorrect data values related to part or external file meta-data should *not* be included here as they are accounted for earlier in this template.

Line 23 – Average time spent acquiring the relevant external bulk data

This is the average amount of labour time per exchange that is spent trying to acquire relevant externally referenced bulk data for which the exchange has failed due to corrupt association between part meta-data and external reference data files.

Line 24 – Average time spent on re-synchronisation of externally referenced bulk data

This is the average amount of labour time per exchange that is spent re-entering relevant meta-data about externally referenced information and/or re-associating the relevant externally referenced bulk data with the correct part meta-data.

Line 27 – Number of exchanges with corrupt product structure associations between part meta-data

This is the average number of PDM files that have failed to complete the exchange process in a given month due to corrupt product structure associations between part meta-data. This is the explicit part assembly relationship structure, often formatted as an explicit bill of material. Incorrect data values related to individual part information meta-data should *not* be included here as they are accounted for earlier in this template.

Line 28 – Average time spent on product structure information re-entry

This is the average amount of labour time per exchange that is spent acquiring and re-entering relevant product structure information among the part meta-data information elements.

Line 29 – Average time lost using bad product structures

This is the average amount of time per exchange that is lost as a result of work done on the basis of imported PDM models that contain corrupt product structure associations between part meta-data. This is work done prior to discovery of corrupt product structures that must be re-done in light of the correct structure. It includes both design as well as manufacturing re-work.

Line 30 – Average time spent resolving inconsistent PDM and CAD structures

This is the average amount of time per exchange that is lost as a result of work done on the basis of imported PDM models that contain product structure associations between part meta-data that is inconsistent with the geometric model relationships that exist between part shape representations. Although the part shape representation is typically external bulk data to the PDM, the structure relationships among external shapes are often in scope for PDM. This is particularly important for the digital mock-up application.

Line 31 – Average time spent on external geometric model structure information re-entry

This is the average amount of labour time per exchange that is spent acquiring and re-entering relevant external geometric model structure information among the external part shape meta-data information elements.

F.1.37 Costs of poor data organisation and classification

Line 36 – Average time spent locating properly categorised information within the PDM system

This is the average amount of labour time per month that is spent trying to locate required information within the PDM system where the search for information was impeded by an inadequate or unexpected data organisation/classification scheme within the PDM system.

Line 37 – Average time spent locating improperly categorised information within the PDM system

This is the average amount of labour time per month that is spent trying to locate required information within the PDM system where the search for information was impeded by an incorrect categorisation of the required information.

Line 38 – Average time spent locating information within the computer file system

This is the average amount of labour time per month that is spent trying to locate required information identified by the PDM system where the search for information was impeded by an inadequate or unexpected directory structure within the computer file system referenced by the PDM meta-data.

F.1.38 Costs of poor data packaging

Line 41 – Number of exchanges involving a technical data package

This is the average number of technical data packages that are exchanged in a given month. In this context, a technical data package typically consists of a PDM meta-data exchange file packaged together with a set of bulk data files that are externally referenced from the PDM data.

Line 42 – Average time spent searching for referenced bulk data files not present in a technical data package

This is the average amount of labour time per exchange that is spent trying to locate relevant externally referenced bulk data for PDM models where the external reference data files are not present in the imported technical data package.

Line 43 – Average time spent acquiring referenced bulk data files not present in an imported technical data package

This is the average amount of labour time per exchange that is spent actually acquiring (after locating) the relevant externally referenced bulk data for an imported technical data package where the external reference data files are not present in the technical data package.

F.1.39 Costs of PDM data redundancy

Line 46 – Number of redundant PDM data models

This is the number of different data models that exist within the scope of your PDM and data management systems that contain information that is also (redundantly) contained and/or managed within another data model/system.

Line 47 – Average time spent ensuring consistency across all PDM and data management systems

This is the average amount of labour time per month that is spent trying to maintain consistent data values across redundant information that is managed by more than one PDM or data management system.

Line 48 – Average time spent on redundant data value re-entry

This is the average amount of labour time per month that is spent re-entering consistent data values into data repositories that contain redundant information maintained by more than one PDM or data management system.

Line 49 – Average time spent translating between redundant data repositories

This is the average amount of labour time per month that is spent translating data values between data repositories that contain redundant information maintained by more than one PDM or data management system.

Line 50 – Average time lost using inconsistent data values

This is the average amount of time per month that is lost as a result of work done on the basis of PDM information that is corrupt due to lack of consistency between data repositories that contain redundant information maintained by more than one PDM or data management system. It includes both design as well as manufacturing re-work.

F.1.40 Costs due to lack of integration among PDM and materials and parts libraries and catalogues

Line 53 – Number of part libraries and catalogues

This is the number of different standard part libraries and/or catalogues that are maintained or accessed across the scope of your PDM and data management systems.

Line 54 – Average time lost using outdated part library or catalogue information

This is the average amount of time per month that is lost as a result of work done on the basis of PDM information that is out of date due to a lack of timely updates to the library/catalogue information.

Line 55 – Average time lost in failed attempts to access standard part library or catalogue information

This is the average amount of labour time per month that is lost trying but failing to locate standard part information that is managed by a standard part library or catalogue system.

Line 56 – Average time spent ensuring consistency across part libraries or catalogues

This is the average amount of labour time per month that is spent trying to maintain consistent data values across redundant information that is managed by more than one standard part library or catalogue system.

F.1.41 Costs of poor integration between manufacturing system and PDM

Line 59 – Average time spent on BOM data value re-entry from PDM to MRP systems

This is the average amount of labour time per month that is spent re-entering consistent data values for the part structure/bill of material information from a PDM system into MRP systems.

Line 60 – Average time spent on process specification data value re-entry from PDM to manufacturing systems

This is the average amount of labour time per month that is spent re-entering consistent data values for the part process specification information from a PDM system into manufacturing systems.

Line 61 – Average time spent locating warranty service part numbers from PDM system

This is the average amount of labour time per month that is spent locating part meta-data in the PDM system that identifies the as-built identification of a product for maintenance and product life cycle support.

F.1.42 Costs due to security / access problems

Line 64 – Average time spent working around PDM system security

This is the average amount of labour time per month that is spent working around system security and permissions to gain access to PDM information.

Line 65 – Average time spent searching for information that is not accessible due to PDM system security

This is the average amount of labour time per month that is spent searching in other systems for information that is inaccessible from within the PDM system due to security and/or system permissions.

F.1.43 Potential cost savings

Line 69 – Loss of business

This is the average annual cost of business lost due to PDM data quality issues, either current business or future business.

Line 70 – Other costs

These are any additional average annual costs related to PDM data quality issues that might be company-specific or not mentioned in this template.

To calculate costs for addressing product data quality problems related to PDM data, see template F.1.44.

F.1.44 Annual PDM Product Data Quality Costs

	1	Hourly labour cost (employee's rate + overhead)	**1**	**$**		
Cost of failed data exchanges involving PDM **(calculated on a monthly basis)**	**2**	Average number of data exchanges involving PDM per month	**2**			
	3	Monthly number of PDM data exchanges that fail due to errors in the data values	**3**			
	4	Time per exchange spent identifying bad data values	**4**	**hrs**		
	5	Time per exchange spent acquiring correct replacement values	**5**	**hrs**		
	6	Time per exchange spent in data value re-entry/repair	**6**	**hrs**		
	7	Time per exchange lost working with bad data values	**7**	**hrs**		
	8	Add lines 4 through 7. This is the total average time per file spent addressing PDM data errors.	**8**	**hrs**		
	9	Multiply line 8 by line 3. This is the average time per exchange spent as a result of errors in data values.	**9**	**hrs**		
ECN	**10**	Monthly number of times the association between part data and ECN data are missing	**10**			
	11	Time per exchange spent locating relevant ECN data	**11**	**hrs**		
	12	Time per exchange spent re-entering or fixing ECN data	**12**	**hrs**		
	13	Time per exchange wasted due to working with outdated versions of the part data	**13**	**hrs**		
	14	Time per exchange spent re-engineering a new design for reasons captured in earlier data	**14**	**hrs**		
	15	Add lines 11 through 14. This is the total average time per file spent addressing part-data/ECN-data association errors.	**15**	**hrs**		
	16	Multiply line 15 by line 10. This is the average cost per exchange due to part-data/ECN-data association errors.	**16**	**hrs**		
Authorisation	**17**	Monthly number of times the association between part data and authorisation data are missing	**17**			
	18	Time spent per exchange verifying a required approval or sign-off	**18**	**hrs**		
	19	Time per exchange spent re-entering or re-associating authorisation data	**19**	**hrs**		
	20	Add lines 18 through 19. This is the total average time per file spent addressing part-data/approval association errors.	**20**	**hrs**		
	21	Multiply line 20 by line 17. This is the average cost per exchange due to part-data/approval association errors.	**21**	**hrs**		
External reference	**22**	Monthly number of times the association between part data and external bulk data are missing	**22**			
	23	Time per exchange spent acquiring a copy of the externally referenced data	**23**	**hrs**		
	24	Time per exchange spent re-synchronising PDM system with external data system	**24**	**hrs**		
	25	Add lines 23 through 24. This is the total average time per file spent addressing part-data/bulk-data association errors.	**25**	**hrs**		
	26	Multiply line 25 by line 22. This is the average cost per exchange due to part-data/bulk-data association errors.	**26**	**hrs**		
Product structure	**27**	Monthly number of PDM data exchanges that fail due to missing product structure associations among part data	**27**			
	28	Time per exchange spent recreating the product structure in the PDM system	**28**	**hrs**		
	29	Time per exchange spent working on incorrect or ineffective product structures	**29**	**hrs**		
	30	Time per exchange spent addressing inconsistencies between PDM and CAD representations of product structure	**30**	**hrs**		
	31	Time per exchange spent due to lost geometric relationships between parts in PDM	**31**	**hrs**		
	32	Add lines 28 through 31. This is the total average time per file spent addressing PDM data errors.	**32**	**hrs**		
	33	Multiply line 32 by line 27. This is the average cost per exchange due to error in data values.	**33**	**hrs**		
	34	Add lines 9, 16, 21, 26, and 33. This is your monthly time spent per exchange on corrupted data.	**34**	**hrs**		
	35	Multiply line 34 by line 1 and then by the number 12. This is your annual cost for failed exchanges due to PDM problems.			**35**	**$**

Costs of a poor data organisation/classification system **(calculated on a monthly basis)**	**36**	Time per month spent trying to locate data due to inadequate classification/organisation, including data in unexpected location	**36**	**hrs**		
	37	Time per month spent due to inconsistent/incorrect naming or classification of data elements	**37**	**hrs**		
	38	Time per month spent trying to locate data due to an inadequate, inconsistent, or non-intuitive directory structure	**38**	**hrs**		
	39	Add lines 36 through 38. This is the total average time per file spent addressing PDM data errors.	**39**	**hrs**		
	40	Multiply line 39 by line 1 and then by the number 12. This is the average annual cost due to a poor data organisation/classification system.			**40**	**$**
Costs of poor data packaging **(calculated on a monthly basis)**	**41**	Average monthly number of technical data packages exchanged	**41**			
	42	Average time spent for each technical data package trying to access files referenced but not present	**42**	**hrs**		
	43	Average time spent for each technical data package acquiring files that were not received as part of the package	**43**	**hrs**		
	44	Add lines 42 and 43. This is the total average time per technical data package spent on associated file problems.	**44**	**hrs**		
	45	Multiply line 44 by line 1 and then by the number 12. This is the average annual cost due to poor packaging of data.			**45**	**$**
Costs of redundant PDM data **(calculated on a monthly basis)**	**46**	Number of redundant data models across the PDM and data management systems	**46**			
	47	Average monthly time spent ensuring consistency across redundant data models	**47**	**hrs**		
	48	Average monthly time spent manually re-entering data into redundant data repositories	**48**	**hrs**		
	49	Average monthly time spent translating data between redundant data repositories	**49**	**hrs**		
	50	Average monthly time spent working with the wrong data because it was not kept consistent across repositories	**50**	**hrs**		
	51	Add lines 47 through 50. This is the monthly average time spent due to redundant data.	**51**	**hrs**		
	52	Multiply line 51 by line 1 and then by the number 12. This is the average annual cost due to redundant data.			**52**	**$**
Costs due to lack of integration among PDM and libraries and catalogues **(calculated on a monthly basis)**	**53**	Number of libraries and catalogues maintained across the company	**53**			
	54	Average monthly time lost due to libraries or catalogues not being updated	**54**	**hrs**		
	55	Average monthly time spent trying but failing to access libraries or catalogues	**55**	**hrs**		
	56	Average monthly time spent ensuring consistency across multiple libraries or catalogues	**56**	**hrs**		
	57	Add lines 54 through 56. This is the monthly average time spent due to lack of library/catalogue integration.	**57**	**hrs**		
	58	Multiply line 57 by line 1 and then by the number 12. This is the average annual labour cost due to lack of library/catalogue integration.			**58**	**$**
Costs of poor integration between mfg. system and PDM **(calculated on a monthly basis)**	**59**	Average monthly time spent re-entering BOM information into MRP system	**59**	**hrs**		
	60	Average monthly time spent re-entering lost process specification information	**60**	**hrs**		
	61	Average monthly time spent finding actual service part numbers for warranty	**61**	**hrs**		
	62	Add lines 59 through 61. This is the monthly average time spent due to poor integration of manufacturing and PDM	**62**	**hrs**		
	63	Multiply line 62 by line 1 and then by the number 12. This is the average annual cost due to poor integration of manufacturing and PDM.			**63**	**$**

Costs due to security/ access problems (calculated on a monthly basis)	**64**	Average monthly time spent working around system permission problems to access needed data	**64**	**hrs**		
	65	Average monthly time spent looking elsewhere for data that were not available due to permission/access problems	**65**	**hrs**		
	66	Add lines 64 and 65. This is the monthly average time spent addressing access/security problems.	**66**	**hrs**		
	67	Multiply line 66 by line 1 and then by the number 12. This is the average annual cost due to problems from security systems and access limitations.			**67**	**$**
Potential cost savings	**68**	Add lines 35, 40, 45, 52, 58, 63, and 67. This is a subtotal of your annual costs due to PDM exchange-related data quality problems.	**68**	**$**		
	69	Annual loss of business due to PDM data quality issues	**69**	**$**		
	70	Other annual costs related to PDM data quality	**70**	**$**		
	71	Add lines 68 through 70. These are your total potential annual cost savings			**71**	**$**

F.1.45 Start-up and Annual Costs of Improving Product Data Quality

Product data quality improvement does not come free. It is important to estimate the costs associated with implementing a change intended to improve data quality. The template below F.1.48. lists a basic set of costs to consider. There is only one cost-of-improvement template because the same kinds of costs apply regardless of the particular focus of a quality improvement effort.

While many costs are one-time costs, there are also significant recurring costs as well. Therefore, this costs section is divided into the start-up and subsequent annual costs to address product data quality.

The following line-by-line instructions provide information on how to fill out this template. Line-by-line instructions are provided for all direct data entry lines. Lines calculated from data already in the template are not described here.

F.1.46 Cost to Implement

These are the costs at the beginning of a change process. In general, they are the first-year costs. Ongoing (maintenance) costs are captured in the next template section.

Line 1 – Labour cost to determine requirements

Changes should not be implemented on an ad hoc basis. Requirements should be determined for any changes before determining the details of the change. For something as complicated as CAD data quality, this is not a quick, easy process. Estimating the necessary cost (in labour) is important to doing it properly.

Line 2 – Labour cost to determine roles and responsibilities

Based on the requirements, determine the appropriate roles and responsibilities for carrying out the change. Estimate the labour cost it will take to do that. Be sure to take into account the costs involved in significantly changing people’s roles.

Line 3 – Cost to address human resource issues

Any significant changes in roles and responsibilities can have broad effects on job descriptions. Consider the effects of changes in job descriptions and responsibilities in such areas as pay classifications and union contract issues.

Line 4 – Labour cost to establish standards and metrics

Establish standards that the solution will be expected to meet or support. Similarly, decide on how to measure the success of the solution. Estimate the labour cost it will take to create these standards and metrics. Most of the labour cost will be spent on people participating in the committees that establish the standards and metrics.

Line 5 – Establish a process for change

Ensuring an effective change requires establishing a documented process for making that change. Estimate the cost of labour for this process planning.

Line 6 – Cost to establish and conduct a training program

Regardless of the change that is undertaken, personnel will need to be trained in how to take advantage of the change. The cost will depend on how many people need to be trained and the complexity of the information they need to learn. Estimate the cost of training for this change, including training development and labour time for people taking the training.

Line 7 – Cost of extra time required for the learning curve

People will always need time to adjust to any new way of doing things, whether it involves technology or not. The reduction in productivity during this period should be taken into account as an increased cost. This cost is in addition to the cost of time spent in formal training.

Line 8 – Cost of technology and software tools required

Estimate the cost of acquired technology needed to complete the improved approach. Depending on the particular situation, technology may be a major or a minor part of the overall implementation cost.

Line 9 – Cost of adverse relations with customer

Implementing a change is likely to disrupt normal activities and may affect customer relations. Estimate the potential cost of annoying or imposing requirements on the customer in the course of undertaking the improvement. Such costs will be realised through lost sales or less flexibility on pricing.

Line 10 – Other costs related to implementation

Enter on this line any other costs that can be predicted will arise in the process of implementing improvements.

F.1.47 Cost to Maintain

These are the annual costs that occur once a changed process has been completed. In general, they apply after the first year of the change.

Line 12 – Labour cost to update requirements

Changes should not be made and then assumed to have solved the problem permanently. Review on a regular basis the labour requirements and costs established for the original changes. Any revisions should contribute to the maintenance process. You need to estimate the necessary cost (in labour) to properly conduct the requirements review.

Line 13 – Labour costs to update roles and responsibilities

Based on the (potentially revised) requirements, you need to review the assigned roles and responsibilities that resulted from the original and any subsequent change processes. Estimate the labour cost it will take to conduct that review and revision process.

Line 14 – Cost to address new human resource issues

Review the human resource issues raised in the original change process and consider any new issues that might have surfaced related to the changed system. For the template, estimate the cost of conducting the review and implementing any changes that result.

Line 15 – Labour cost to update standards and metrics

You should review the standards and metrics that were developed for the original and subsequent changes. Estimate the labour cost it will take to conduct the review and update. Most of the labour cost will be spent on people participating in the committees that establish the standards and metrics.

Line 16 – Labour cost to establish an upgrade process

The result of the reviews described in lines 12-15 may result in the need for further change. As in the initial change, ensuring an effective change requires establishing a documented process for making that change. Estimate the cost of this process planning.

Line 17 – Cost to maintain the training program

No matter how effective the initial training, modifications and updates to the systems as well as new employees being brought into the system will generate training requirements. Estimate the cost to review and update the training and to provide it to the appropriate personnel. The cost will depend on how many people need to be trained, to what level, and the complexity of the information they need to learn. Estimate the cost of recurrent and continuing need for initial training, including training development and labour time for people taking the training.

Line 18 – Cost of technology and software maintenance

Estimate the annual cost of maintaining the technology you will have acquired for the improved approach. Depending on the particular situation and technologies involved, maintenance may be a major or a minor part of the overall maintenance cost.

Line 19 – Other costs related to maintenance

These are any other costs that you foresee in the process of maintaining the improvements.

F.1.48 Costs of Improving Product Data Quality Template

Cost to implement (start-up costs)	**1**	Labour cost to determine requirements	**1**	**$**		
	2	Labour cost to determine roles and responsibilities	**2**	**$**		
	3	Cost to address human resource issues	**3**	**$**		
	4	Labour cost for committee participation to establish standards and metrics	**4**	**$**		
	5	Labour cost to establish a process for change	**5**	**$**		
	6	Cost to establish and conduct a training program	**6**	**$**		
	7	Cost of extra time required for the learning curve	**7**	**$**		
	8	Cost of technology and software tools required	**8**	**$**		
	9	Cost of adverse relations with customer	**9**	**$**		
	10	Other costs related to implementation	**10**	**$**		
	11	Add lines 1 through 10. This is your total cost to implement.			**11**	**$**
Cost to maintain (annual costs)	**12**	Labour cost to update requirements	**12**	**$**		
	13	Labour cost to update roles and responsibilities	**13**	**$**		
	14	Cost to address new human resource issues	**14**	**$**		
	15	Labour cost for committee participation to maintain standards and metrics	**15**	**$**		
	16	Labour cost to establish upgrade process	**16**	**$**		
	17	Cost to maintain a training program	**17**	**$**		
	18	Cost of technology and software maintenance	**18**	**$**		
	19	Other costs related to maintenance	**19**	**$**		
	20	Add lines 12 through 19. This is your total annual cost to maintain.			**20**	**$**

F.1.49 Product Data Quality Cost Summary

This template F.1.54. brings the "bottom lines" of all the other templates together into a single place. It therefore presents the costs that result from product data quality problems as well as the potential costs of improving product data quality.

The following line-by-line instructions provide information on how to fill out this template.

F.1.50 Supporting Information

Line 1 – Annual sales

This is the annual sales for the organisation (company, division, business unit, etc.) for which the PDQ costs have been calculated. This value is used to convert the PDQ costs and savings into a percentage of annual sales.

F.1.51 Costs Due to PDQ Problems

This part of the template gathers together the various costs of product data quality problems into a single place.

Line 2 – CAD-related PDQ costs

These are the costs from the CAD product data quality template F.1.7 , as summarised in line 54 of that template.

Line 4 – CAM-related PDQ costs

These are the costs from the CAM product data quality template F.1.15 , as summarised in line 54 of that template.

Line 6 – CAE-related PDQ costs

These are the costs from the CAE product data quality template F.1.21. , as summarised in line 36 of that template.

Line 8 – Prototyping-related PDQ costs

These are the costs from the prototyping product data quality template F.1.27. , as summarised in line 36 of that template.

Line 10 – Digital-mock-up-related PDQ costs

These are the costs from the digital mock-up product data quality template F.1.33. , as summarised in line 36 of the template.

Line 12 – PDM costs due to PDQ problems

These are the costs from the PDM product data quality template F.1.44. , as summarised in line 71of that template.

Line 14 – Other costs due to PDQ problems

These are any other annualised costs from product data quality problems.

F.1.52 Costs to Improve Product Data Quality

Line 15 – First-year costs

This part of the template brings forward the implementation and maintenance costs for the short-term (initial implementation) as summarised in line 11 in the implementation template F.1.48 .

Line 17 – Subsequent year costs

Improvement projects are typically not one-shot operations. To maintain an improvement involves spending additional the time and effort to make sure the data are right. This line brings forward the subsequent year costs for maintaining the new approach, as summarised in line 20 of the cost of implementation template F.1.48 .

F.1.53 Potential Savings

Line 19 – Net first-year savings

These are the annual savings in the year during which a process change has been undertaken. In general, they apply after the first year.

Line 21 – Net subsequent-year savings

These are the annual net savings that will accrue during years after the initial year of changed approaches.

F.1.54 Product Data Quality Cost Summary

	1	Annual sales of organisation being analysed, used below	1	$		
Costs due to PDQ problems	2	CAD-related PDQ costs (Line 54 of CAD costs template)	2	$		
	3	Divide line 2 by line 1 and multiply by 100. This is your CAD-related cost as a percentage of your overall sales.	3	%		
	4	CAM-related PDQ costs (Line 54 of CAM costs template)	4	$		
	5	Divide line 4 by line 1 and multiply by 100. This is your CAM-related cost as a percentage of your overall sales.	5	%		
	6	CAE-related PDQ costs (Line 36 of CAE costs template)	6	$		
	7	Divide line 6 by line 1 and multiply by 100. This is your CAE-related cost as a percentage of your overall sales.	7	%		
	8	Prototyping-related PDQ costs (Line 36 of prototype costs template)	8	$		
	9	Divide line 8 by line 1 and multiply by 100. This is your prototype-related cost as a percentage of your overall sales.	9	%		
	10	Digital mock-up-related PDQ costs (Line 36 of digital-mock-up costs template)	10	$		
	11	Divide line 10 by line 1 and multiply by 100. This is your digital-mock-up-related cost as a percentage of your overall sales.	11	%		
	12	PDM-related PDQ costs (Line 71 of PDM costs template)	12	$		
	13	Divide line 12 by line 1 and multiply by 100. This is your digital-mock-up-related cost as a percentage of your overall sales.	13	%		
	14	Other PDQ costs identified but not documented in the provided template	14	$		
	15	Add lines 2, 4, 6, 8, 10, 12, and 14. This is your total cost due to PDQ problems.			15	$
	16	Divide line 15 by line 1 and multiply by 100. This is your overall PDQ-related cost as a percentage of your overall sales.			16	%
Costs to implement improvements	17	First-year costs to implement improvements (Line 11 of implementation cost template)			17	$
	18	Divide line 17 by line 1 and multiply by 100. This is your first-year cost to implement PDQ improvements as a percentage of your overall sales.			18	%
	19	Subsequent-year costs to maintain and continue improvements (Line 20 of implementation cost template)			19	$
	20	Divide line 19 by line 1 and multiply by 100. This is your subsequent-year cost to implement PDQ improvements as a percentage of your overall sales.			20	%
Potential savings	21	If line 15 is more than line 17, subtract line 17 from line 15. These are your **net potential savings in the first year after installation of improvements.**			21	$
	22	Divide line 21 by line 1 and multiply by 100. This is your net potential first-year savings from implementing PDQ improvements as a percentage of your overall sales.			22	%
	23	If line 15 is more than line 19, subtract line 19 from line 15. These are your **net potential savings in subsequent years after installation of improvements.**			23	$
	24	Divide line 23 by line 1 and multiply by 100. This is your net potential subsequent-year savings from implementing PDQ improvements as a percentage of your overall sales.			24	%

Attachment G - Revision request

Please use this format for submitting all comments and issues (in English) on the SASIG PDQ guideline to the keeper of the issues log (contact information below).

Location in document (section/paragraph number):

Description of issue (Describe what needs to be changed, including a justification)**:**

Proposed solution (Give your proposed solution. Include new wording, recommended document organisation, etc., as appropriate):

Issues Log Keeper: Louis Fort

Renault

Department 12250 – API: TCR PLU 110

Technocentre Renault

1 avenue du Golf

F78288 Guyancourt Cedex

louis.fort@renault.com

Attachment H – XML Schema for the quality stamp

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:element name="Quality-Stamp">
    <xs:complexType>
        <xs:sequence>
        <xs:element name="Part-Information">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Part">
                  <xs:complexType>
                  <xs:attribute name="Name" type="xs:string" use="required"/>
                  <xs:attribute name="Revision" type="xs:string" use="required"/>
                  <xs:attribute name="Size_KB" type="xs:positiveInteger" use="required"/>
                </xs:complexType>
              </xs:element>
              <xs:element name="CAD-System">
                <xs:complexType>
                  <xs:attribute name="Name" type="xs:string" use="required"/>
                  <xs:attribute name="Release" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
              <xs:element name="Last-Save">
                <xs:complexType>
                  <xs:attribute name="Date" type="xs:date" use="required"/>
                  <xs:attribute name="Time" type="xs:time" use="required"/>
                </xs:complexType>
              </xs:element>
              <xs:element name="Version">
                <xs:complexType>
                  <xs:attribute name="Information1" type="xs:string"/>
                  <xs:attribute name="Information2" type="xs:string"/>
                  <xs:attribute name="Information3" type="xs:string"/>
                </xs:complexType>
              </xs:element>
              <xs:element name="Owner">
                <xs:complexType>
                  <xs:attribute name="Organisation" type="xs:string"/>
                  <xs:attribute name="Department" type="xs:string"/>
                  <xs:attribute name="UserID" type="xs:string"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="Check-Environment">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Check-CAD-System">
                <xs:complexType>
                  <xs:attribute name="Name" type="xs:string"/>
                  <xs:attribute name="Release" type="xs:string"/>
                </xs:complexType>
              </xs:element>
```

```
                <xs:element name="Check-Tool">
                  <xs:complexType>
                    <xs:attribute name="Name" type="xs:string" use="required"/>
                    <xs:attribute name="Release" type="xs:string" use="required"/>
                    <xs:attribute name="Check-Profile" type="xs:string" use="required"/>
                    <xs:attribute name="SASIGVersion" type="xs:string" use="required"/>
                  </xs:complexType>
                </xs:element>
                <xs:element name="Check-Date-Time">
                  <xs:complexType>
                    <xs:attribute name="Date" type="xs:date" use="required"/>
                    <xs:attribute name="Time" type="xs:time" use="required"/>
                  </xs:complexType>
                </xs:element>
                <xs:element name="Operating-System">
                  <xs:complexType>
                    <xs:attribute name="Type" type="xs:string" use="required"/>
                    <xs:attribute name="Version" type="xs:string" use="required"/>
                  </xs:complexType>
                </xs:element>
                <xs:element name="Check-Performed">
                  <xs:complexType>
                    <xs:attribute name="Organisation" type="xs:string" use="required"/>
                    <xs:attribute name="Department" type="xs:string" use="required"/>
                    <xs:attribute name="UserID" type="xs:string" use="required"/>
                  </xs:complexType>
                </xs:element>
                <xs:element name="Responsible">
                  <xs:complexType>
                    <xs:attribute name="Organisation" type="xs:string"/>
                    <xs:attribute name="Department" type="xs:string"/>
                    <xs:attribute name="UserID" type="xs:string"/>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
              <xs:attribute name="Checksum" type="xs:string"/>
            </xs:complexType>
          </xs:element>
          <xs:element name="Check-Result">
            <xs:complexType>
              <xs:sequence maxOccurs="unbounded">
                <xs:element name="Criteria">
                  <xs:complexType>
                    <xs:attribute name="SASIG-PDQ-Code" type="xs:string" use="required"/>
                    <xs:attribute name="Internal-Code" type="xs:string"/>
                    <xs:attribute name="Name" type="xs:string"/>
                    <xs:attribute name="Parameter1" type="xs:string"/>
                    <xs:attribute name="Parameter1Unit" type="xs:string"/>
                    <xs:attribute name="Parameter2" type="xs:string"/>
                    <xs:attribute name="Parameter2Unit" type="xs:string"/>
                    <xs:attribute name="Parameter3" type="xs:string"/>
                    <xs:attribute name="Parameter3Unit" type="xs:string"/>
                    <xs:attribute name="Number-Checked-Entities" type="xs:nonNegativeInteger"
use="required"/>
                    <xs:attribute name="Number-Violating-Entities" type="xs:nonNegativeInteger"
use="required"/>
                  </xs:complexType>
```

```
          </xs:element>
        </xs:sequence>
        <xs:attribute name="Quality-Value" type="xs:int"/>
        <xs:attribute name="Number-Total-Entities" type="xs:nonNegativeInteger" use="required"/>
        <xs:attribute name="Number-Checked-Entities" type="xs:nonNegativeInteger" use="required"/>
        <xs:attribute name="Check-Log-File" type="xs:string"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Remark" type="xs:string" minOccurs="0"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>
</xs:schema>
```

Attachment I – XML File Example


```xml
<?xml version="1.0"?>
<Quality-Stamp xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="qualitystamp.xsd">
  <Part-Information>
    <Part Name="SASIG-Example.prt" Revision="001" Size_KB="10752"/>
    <CAD-System Name="MyCAD" Release="V1R02"/>
    <Last-Save Date="2005-05-23" Time="14:10:53"/>
    <Version Information1="" Information2="" Information3=""/>
    <Owner Organisation="Company" Department="Department" UserID="ID"/>
  </Part-Information>
  <Check-Environment Checksum="">
    <Check-CAD-System Name="HisCAD" Release="V2R01"/>
    <Check-Tool Name="Mychecker" Release="2.0" Check-Profile="SASIG-Profile" SASIGVersion="2.1"/>
    <Check-Date-Time Date="2005-05-25" Time="19:00:05"/>
    <Operating-System Type="My OS" Version="15.8"/>
    <Check-Performed Organisation="Company" Department="Department" UserID="ID"/>
    <Responsible Organisation="Company" Department="Department" UserID="ID"/>
  </Check-Environment>
  <Check-Result Quality-Value="74" Number-Total-Entities="1500" Number-Checked-Entities="1400" Check-
Log-File="SASIG-Example.result">
    <Criteria SASIG-PDQ-Code="G-SU-LG" Internal-Code="MCE-S09" Name="Large surface gap"
Parameter1="0.02" Parameter1Unit="mm" Parameter2="" Parameter2Unit="" Parameter3="" Parameter3Unit=""
Number-Checked-Entities="15" Number-Violating-Entities="3"/>
    <Criteria SASIG-PDQ-Code="O-CM-IN" Internal-Code="OCE-M07" Name="Non-standard item name"
Parameter1="No" Parameter1Unit="" Parameter2="" Parameter2Unit="" Parameter3="" Parameter3Unit=""
Number-Checked-Entities="1" Number-Violating-Entities="0"/>
  </Check-Result>
  <Remark>This is an example for the SASIG-PDQ Quality Stamp Version 2.1</Remark>

</Quality-Stamp>
```

INDEX

ICS 25.040.40

Price based on 194 pages