COLIBRI – A COLLABORATIVE DESIGN SYSTEM FOR PRODUCT DATA INTEGRATION

Sven Kleiner, Reiner Anderl and Robert Gräb

Keywords: Product Data Integration, Collaborative Systems, CAx, PDM, EDM

1. Introduction

The increasing pressure towards lower costs and shorter development cycles requires the use of computer aided methods and modern software systems already in early design phases of product development. Today, innovative products are often mechatronic products and therefore include mechanical, hydraulic, pneumatic, electric, controller, and even software components. Thus, CAD systems are required to develop geometry models and digital mock-ups (DMU) and calculations and system simulations have to be performed as soon as practicable.

In early design stages of virtual product development processes, the product models of the involved mechatronic engineering domains are characterized by incomplete information and have to be modified very often. This means that there is a need to integrate all partial models into a general product model and to support the necessary data exchange between different application systems.

2. Interdisciplinary Constraints in Product Development

CAD systems provide parametric and feature based modelling methods, support frequent model changes and are well suited for the use in early design phases. Product models for analysis and simulation are usually built up parametrically by specific software packages for finite element analysis (FEA), multi body system simulation (MBS), controller design (CACD), etc. In order to define and analyse product attributes different product models such as design models, kinematic models, hydraulic models, electrical models, and system models are needed. The sum of the mentioned partial models in product development represent the holistic structure and characteristic of the intended product.

Except for geometry based data transfers (e.g. CAD-FEA, CAD-MBS) there is neither exchange nor integration of data for interdisciplinary product development available. Hence, a new approach was researched, which links product models using constraints between parameters. The integration concept is based on parametric product models, which share their properties using constraints. The virtual product is represented by partial models from different engineering disciplines and constraint models.

Figure 1 shows the concept of interdisciplinary constraints representing relationships between parameters of product models.

3. Constraint based Integration of Product Models

The fundamentals for the development of neutral, parametric information structures for the integration of product models are provided by existing product data models or data models from ongoing development as well as concepts from constraint logic programming [ISO 2001], [Donges et al. 1999], [Frühwirt et al. 1997]. The design of an extended, parametric information model could consider only

few, basic entities of STEP data models, for example units of functionality (UoF) like product_management_data (S1), element_structure (S2), item_defintion_structure (S3) and kinematics (K1). An information model for parameters and constraints was developed because standardisation activities in the area of parametrics are in work and are restricted on relations between geometrical information.
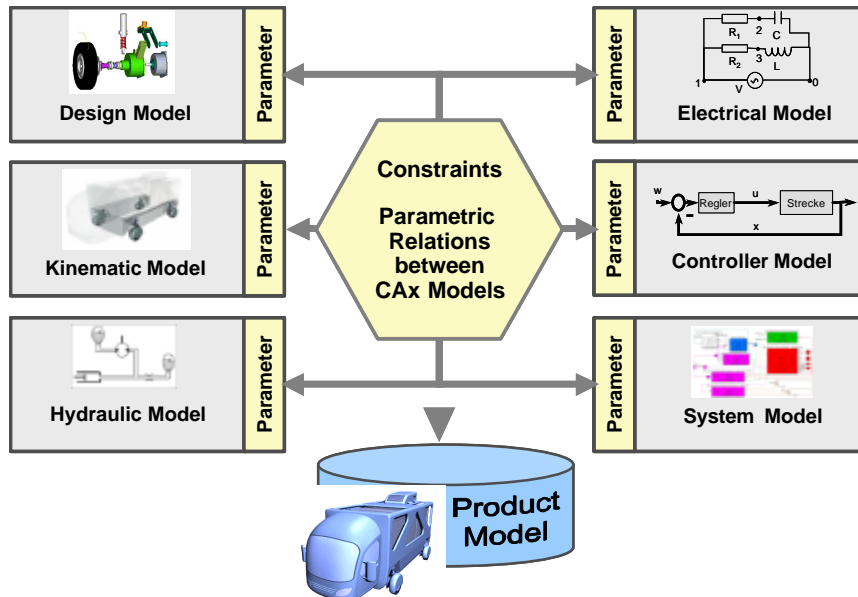

Figure 1. Interdisciplinary Constraints between Product Models

The extended parametric information model was developed using the Unified Modeling Language (UML). The model is containing the class Item, which represents real or virtual objects such as parts, assemblies, and models. Every object Item has a version (class ItemVersion) and specific views (class DesignDisciplineItemDefintion). A view is relevant for the requirements of one or more life cycle stages and application domains and collects product data of the Item and ItemVersion object. The extension of STEP product data models considers the inclusion of general product characteristics (class Property), attributes (class Parameter) and restricted relationships (class Constraint). The developed information model is based on the integration of independent CAx models using its parameters. The links between CAx models are implemented using the class Constraint, which can set parameters of different product models in relationship to each other.

On the one hand a constraint restricts at least one parameter and on the other hand a parameter may be restricted by several constraints, which are building a constraint net. Different types of constraints are implemented in sub classes in order to characterise the relationship between parameters in detail. For example, equal constraints, equal except unit constraints, lower and higher constraints, approximately constraints, and interval constraints are predefined constraints. Constraint nets and hierarchical constraints are represented by the sub class CompositeConstraint.

The super class Constraint offers methods such as check() and solve(). The method check() computes the current state of a constraint, which is represented by a boolean data type (constraint fulfilled, constraint not fulfilled). The method solve() asks for a set of driving parameters (input data) and a set of driven parameters (output data) as arguments. The implementation of a solve() method depends on the constraint type. For example, the solve() method of the class EqualConstraint sets a given parameters equal to another parameter including its unit. While predefined constraints can be solved easily, user defined and composite constraints require algorithms like Newton Method or optimisation routines.

Figure 2 illustrates the main classes and the structure of the developed parametric information model. The information model allows an integration using interdisciplinary constraints restricting attributes of product properties represented as parameters, vectors or matrixes.
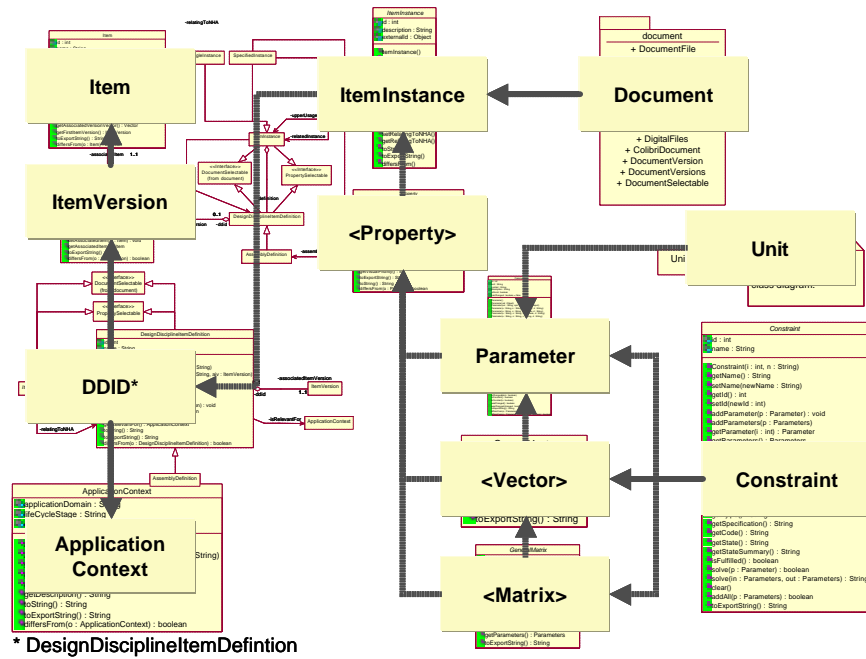
Figure 2. Schematic structure of the extended, parametric information model

4. The Constraint Linking Bridge

The constraint based parametric integration offers an alternative solution compared to unidirectional process chains or file-based data exchange procedures using neutral data formats. Model structures and properties could be imported, analysed, and exported by linking different CAx models. Dependencies between parameters of models are specified with constraints. Constraints form with associative references to CAx models parametric integration models [Gräb 2001]. Besides product data exchange complex model transformations or spanning model optimisations are possible. These transformations depend on available constraint types and constraint solver mechanism.

Based on the constraint based integration concept and the described information model, a software system has been developed based on Java, which supports product data integration for the collaborative design of mechatronic products. The software system Colibri (Constraint Linking Bridge) is setting up connections to CAx systems in order to analyse model structures and parameters. Different interfaces to CAD and CAE systems allow the transfer of appropriate model structures and parameters as well as parameter transformation activities. Browsers for models (Model Viewer) and constraints (Constraint Viewer) are used as graphical user interfaces to analyse, specify, check, and solve constraints. The current version of Colibri contains implemented interfaces to the CAD system Pro/ENGINEER, the simulation application system MATRIXx/SystemBUILD as well as to the multi body simulation software system AUTOLEV. Colibri is integrated into a distributed product development environment, which is called Distributed Computing Environment/Distributed File System (DCE/DFS). This infrastructure offers platform independent security and directory services for users and groups, policy and access control management as well as secure file services storing objects in electronic vaults. Figure 3 shows the architecture of the software system Colibri.

5. Collaborative Design of an Integrated Wheel Suspension assisted by Colibri

In different projects between the Department of Computer Integrated Design (DiK) and partners from industry and university, the development of mechatronic products has been analysed and supported. The application of Colibri will be described exemplified by the design of an integrated wheel suspension of an innovative service vehicle, which was originally motivated by a research project and a cooperation of three research teams [Anderl et al. 2000].

A vehicle for the distribution of goods in cities has been designed in order to check interactions

between different engineering disciplines during the product development process. Compared to heavy trucks, the service vehicle is a delivery van tailored to requirements of city traffic, for example low air pollution, flexible steering mechanism, independent wheel suspension. The final physical DMU of the service vehicle, which was animated in a virtual scene, is illustrated in figure 4.
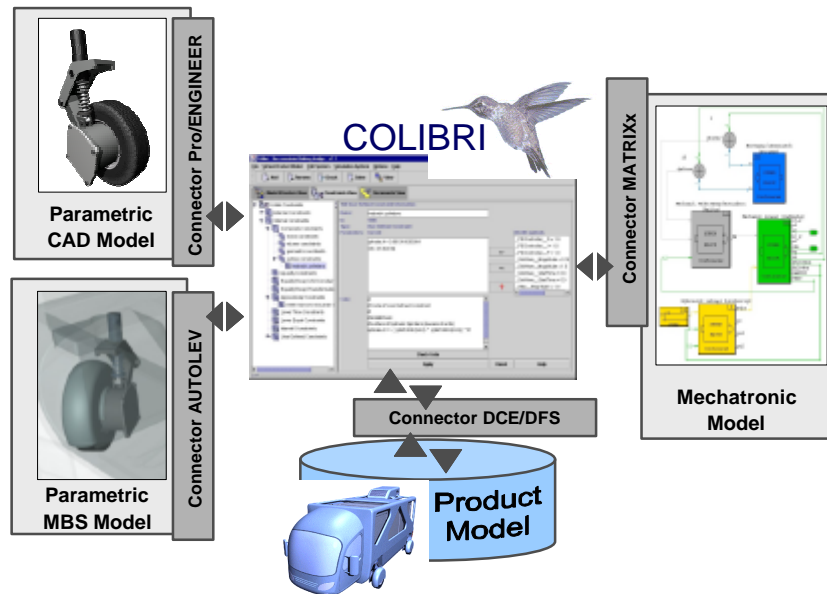

Figure 3. System architecture of Colibri

For the development of mechatronic systems a software environment is needed that allows co-operative design, simulation and optimisation based on an integration platform. Therefore, design methodologies and computer aided tools have been conceived and applied for the development of the wheel suspension at DiK, Mechatronic Labroratory of Paderborn (MLaP), and Heinz Nixdorf Insitute (HNI).
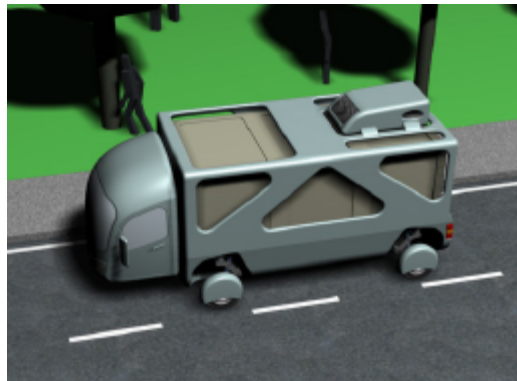

Figure 4. DMU of the innovative service vehicle

New engineering design methods for mechatronic products as well as CAx systems like Pro/ENGINEER, IDEAS, SIMPACK, AUTOLEV, MATRIXx/SystemBUILD, and CAMeL-View were used during development and optimisation without having any hardware prototype of the wheel suspension in early design stages. Students of mechanical engineering, industrial engineering, and electrical engineering were involved to support researchers during design of selected vehicle components and optimisation of system behaviour, for example vertical dynamics of the vehicle.

One specific task was to design the structure and shape of passive components of the active wheel suspension using a CAD system and to model and to analyse dynamical system behaviour of passive components using calculation and simulation software applications.

Parametric CAD models of the wheel module have been designed with the CAD system Pro/ENGINEER, the dynamics of the passive wheel module have been investigated with MBS models using AUTOLEV and mechatronic models of the active wheel module have been created and simulated assisted by MATRIXx/SystemBUILD. Colibri supported the virtual product development according to the constraint based integration of product models. Further, it was used to extract geometrical and technical parameters from one CAx model and to link them to parameters of other CAx models in order to keep data consistent across the domain specific product models. In addition, Engineering Data Management systems (EDM) could be used to support cooperative design and to realise an integrated consistent product structure management (Anderl, Gräb, Kleiner 2001). As a result of the application of Colibri an interdisciplinary integration model of the wheel module was generated.

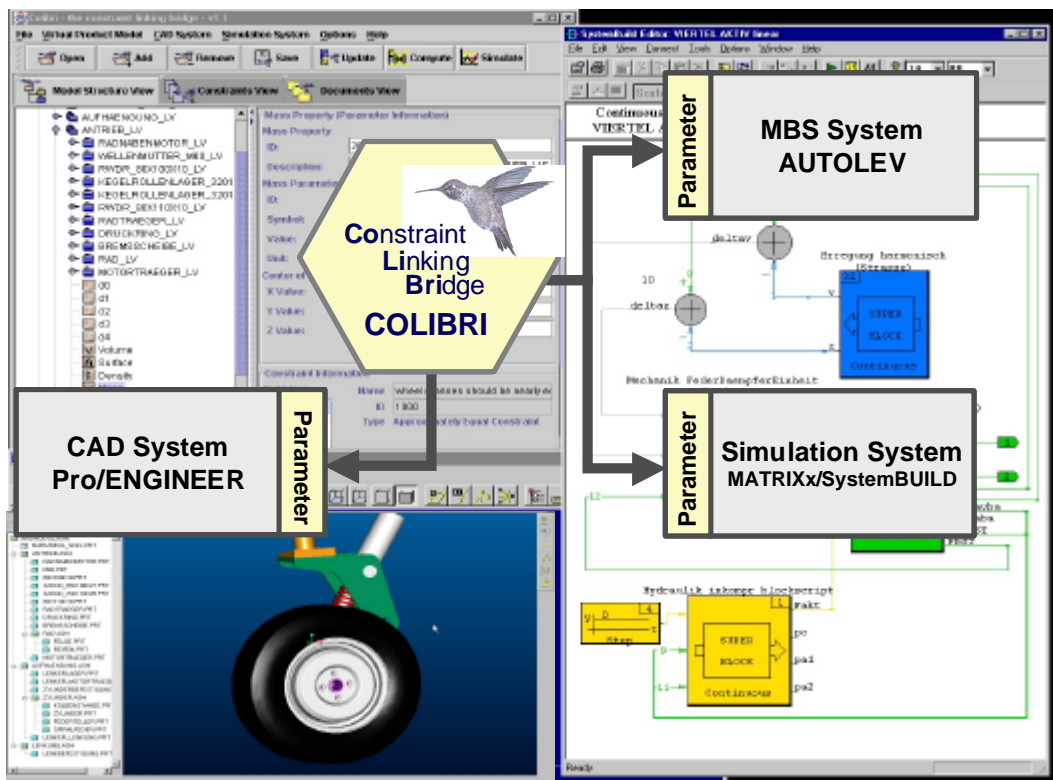Figure 5 shows a screenshot of Colibri and the mentioned CAx systems.

Figure 5. Design of a wheel suspension assisted by Colibri

6. Conclusion and future Work

The implementation of an extended parametric product data model and the application of Colibri have validated the concept of constraint based product data integration. It is possible to integrate different CAx models by just linking them and using constraints to specify their relationships for product development reasons. The software application Colibri, which is in fact a prototype, offers an alternative technology for sharing product data but need to be optimised for daily use based on field test results from engineers.

A central resuming question concerns the enhancement of the constraint based integration concept which goes beyond. The demand for extensions arise from practical experiences and from the potential of constraint management. Interdisciplinary constraints can be defined in the latest program version of Colibri between properties only. Dependencies and relationships between CAx models exist between parameters but also between model nodes and model elements, which need to be considered as well. The following work in research will focus on parametric and structural product data integration and the optimisation of constraint management and user interfaces.

References

Anderl, R., Gräb, R., Kleiner, S., "Integration parametrischer Produktdaten als Grundlage der virtuellen Produktentwicklung". Industrie Management, Vol. 17, pp. 26-30, 2001.

Anderl, R. et al, "Computer Aided Design of Mechatronic Systems exemplified by the Integrated Wheel Suspension of an Inoovative Service Vehicle". Proceedings 1st IFAC-Conference of Mechatronic Systems, VDI-Verlag, Düsseldorf, 2000.

Donges, C., Krastel, M., Anderl, R., "MechaSTEP- STEP Datenmodelle zur Abbildung mechatronischer Systeme", Produktdaten Journal, Heft 1, 1999, pp 30 – 34.

Frühwirt, T., Abdennadher, S., „Constraint-Programmierung", Springer Berlin, 1997.

Gräb, R., „Parametrische Integration von Produktmodellen für die Entwicklung mechatronischer Produkte", D17, Faculty of Mechanical Engineering, Technical University Darmstadt, 2001.

ISO TC 184/SC4/WG12/N940, "ISO/CD 10303-108: Product data representation and exchange. Integrated application resource: Parameterization and constraints for explicit geometric product models", Committee Draft (CD), ISO Switzerland, 2001.

Dipl.-Ing. Sven Kleiner
Technical University Darmstadt
Department of Computer Integrated Design (DiK)
Petersenstr. 30
64287 Darmstadt
Germany
Telephone: +49 60 51 / 16 51 45
Telefax: + 49 60 51 / 16 68 25
Email: kleiner@dik.tu-darmstadt.de