

## **Unterstützung modul-basierter Arbeitsweise mit MEDINA/SDM**

Michael Sattler, T-Systems Enterprise Service GmbH

Thorsten Gerlinger, Kube GmbH Ingenieurbüro

### **Abstract:**

Support of modul based mode of operation within MEDINA/SDM

The current existing functionality in MEDINA/SDM is extended for administration of moduls used for assembly of complex structure. In addition existing process definition tools are configured to integrate check functions as well as visualisation out of the viewer environment.

Using the selection of components, assemblies of the complete structure can be defined, using site specific rules, and stored as assembly definition tables. These tables are enabled to start processes for example to generate input decks – and as basic selection for variations of the structure model.

### **Keywords:**

MEDINA/SDM, Modul, Komponente, Zusammenbau, Verbindung,

## 1 Verwaltung der Module

### 1.1 Navigation

Zur Unterstützung der modul-basierten Arbeitsweise wird das bekannte und in der Praxis bewährte MEDINA/SDM in seiner Grundfunktionalität, ergänzt um wenige spezifische Erweiterungen des Selektionsmechanismus, genutzt.

Aus der Viewer Oberfläche heraus lässt sich sehr einfach durch die vorhandenen Module navigieren. Dabei können Details zu jedem Modul in Form von Bildern und aufbereiteten Meta-Informationen angezeigt werden. Darüber hinaus kann jedes Modul mit Hilfe von MEDINA visualisiert und damit grafisch überprüft werden.

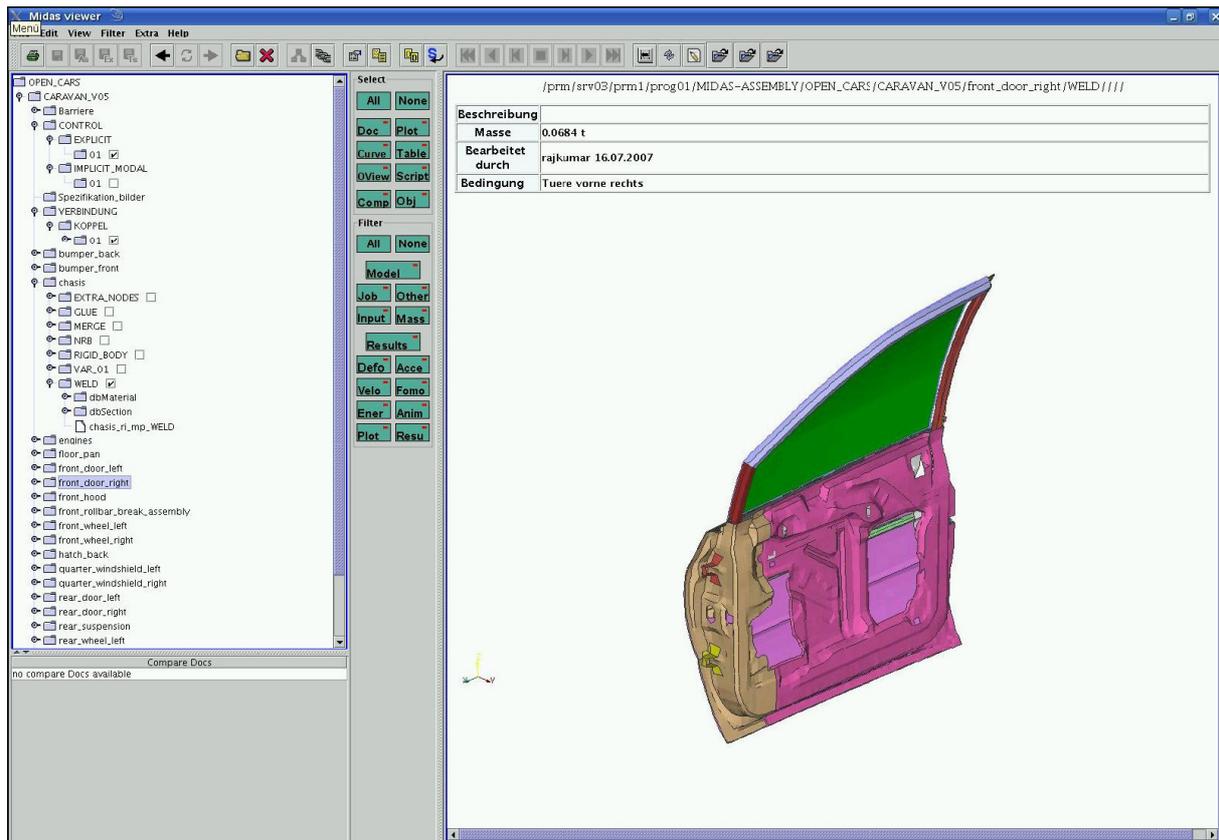


Bild 1: Viewer MEDINA/SDM

### 1.2 Selektion eines Zusammenbaus aus Modulen

Es gibt grundsätzlich eine zweistufige Möglichkeit einen allgemeinen Modellaufbau (auch Teilmodell, mit bzw. ohne Lastfallinformation) durchzuführen.

- Stufe 1: Generierung einer Zusammenbauvorschrift (Assemblydefinition, AD) mit Hilfe von Regeln (Overviewrule, OVR)
- Stufe 2: Ausführen eines vordefinierten Prozess-Templates (Interpreterscript, IS) aus der Zusammenbauvorschrift heraus

Der Ablauf dieser beiden Schritte kann jeweils einzeln und unabhängig voneinander durchgeführt werden, oder auch Schritt 2 automatisch mit der Ausführung an Schritt 1 gekoppelt werden.

Auf dieser Funktionalität kann der Anwender sehr einfach und komfortabel die Selektion der notwendigen Bestandteile des geplanten Zusammenbaus durchführen und damit die in Bild 2 gezeigte Tabelle erzeugen.

**Zusammenbau: car - caravan - dpt - fc - v05:**

| Input Type | FILES    | Layer  | MASS                               | Templates   |
|------------|----------|--------|------------------------------------|---|
| 1          | LOADCASE | 0      |                                    | etypc=J:/TEMPLATES/etypc/car_dpt_fc_d01<br>dyna=J:/TEMPLATES/dyna/template_fc_car_d0<br>midas=J:/TEMPLATES/midas/midbroker_sv.xml<br>loadcase=SV_SV-lett-214.3<br>protocol=J:/PROFILES/coor_sv_BSS_d01.prot<br>workdir=home\dt\cr_T-Systems |
| 2          | BIF      | 1      |                                    |   |
| 3          | BIF      | 100    |                                    |   |
| 4          | BIF      | 210    |                                    |   |
| 5          | BIF      | 220    |                                    |   |
| 6          | BIF      | 230    |                                    |   |
| 7          | BIF      | 240    |                                    |   |
| 8          | INCLUDE  | 760    |                                    |   |
| 9          | BIF      | 770    |                                    |   |
| 10         | PROT     | 100001 |                                    |   |
| 11         | PROT     | 100003 |                                    |   |
| 12         | MASS     | 111111 | 10000001=148.00<br>10000002=311.00 |   |
| 13         | EXE      | 120000 |                                    |   |

Graph: 00 three curves : work\_07\_09\_12 - mm - null - group\_3 - xy\_diagram1  
X-Disp [mm] vs Time [ms]. Curves C1, C2, C3.

Bild 2: Zusammenbautabelle erstellt aus einer Selektion von Modulen

### 1.3 Ablage eines Zusammenbaus und Wiederverwendung

Durch Abspeichern der oben erzeugten Zusammenbautabelle (AD) in der Datenstruktur steht diese für eine spätere Wiederverwendung zur Verfügung.

Diese kann innerhalb der Oberfläche wieder geöffnet werden. Jetzt kann die ursprüngliche Selektion, welche zu diesem Zusammenbau geführt hat, wieder hergestellt und modifiziert werden. Damit lassen sich mit wenigen Aktionen sehr vielfältige Variationen der Simulation durchführen.

### 1.4 Generierung eines rechenfähigen Inputs

Aus der AD heraus lassen sich als kontextsensitive Funktionen Interpreterscripte (IS) starten. Welche Scripte in diesem Kontext vorhanden sind, ist durch die verwendete OVR definiert.

Durch den Ablauf des IS wird der eigentliche Zusammenbau eines rechenfertigen Inputdecks vorgenommen. Dieser kann sowohl durch interaktiven Start von externen Applikationen als auch – wenn ohne Benutzerinteraktion möglich – als komplette Batchprozedur durchgeführt werden.

Dabei können beliebig viele unterschiedliche Prozess Schritte nacheinander durchgeführt werden, wie z.B. Durchführung von Verbindungstechnik und Massenabgleich im Preprozessor, Wandlung der Daten mittels eines Interfaces in ein entsprechendes Solver Datenformat, Integration von „includes“ in das Rechendeck, etc

### 1.5 Überprüfen des Zusammenbaus

Das Gesamtmodell kann in MEDINA sehr komfortabel auf sein Vollständigkeit geprüft werden. Außer dem Fehlen von Bauteilen können auch die ausreichenden Verbindungen zwischen den Bauteilen und die Einhaltung der Nummernräume kontrolliert werden.

Mittels abgestimmter Kontrollrechnungen (Modalanalyse, Anrechnen zur Kontrolle der Rechenfähigkeit und zum Penetrationscheck) kann die Qualität des zusammengebauten Modells geprüft werden.

## 2 Grundsätzliches

### 2.1 Bauteile als standardisierte Einzelkomponenten

Ein Fahrzeug wird aus einer Menge von standardisierten Einzelkomponenten zusammengesetzt. Diese werden nachfolgend mit Bauteile bezeichnet. Jedes dieser Bauteile kann wieder aus mehreren Einzelbauteilen bestehen.

Die Definition von Bauteilen in der Crashberechnung hat das Ziel unterschiedliche Bauteile einfach miteinander kombinieren zu können.

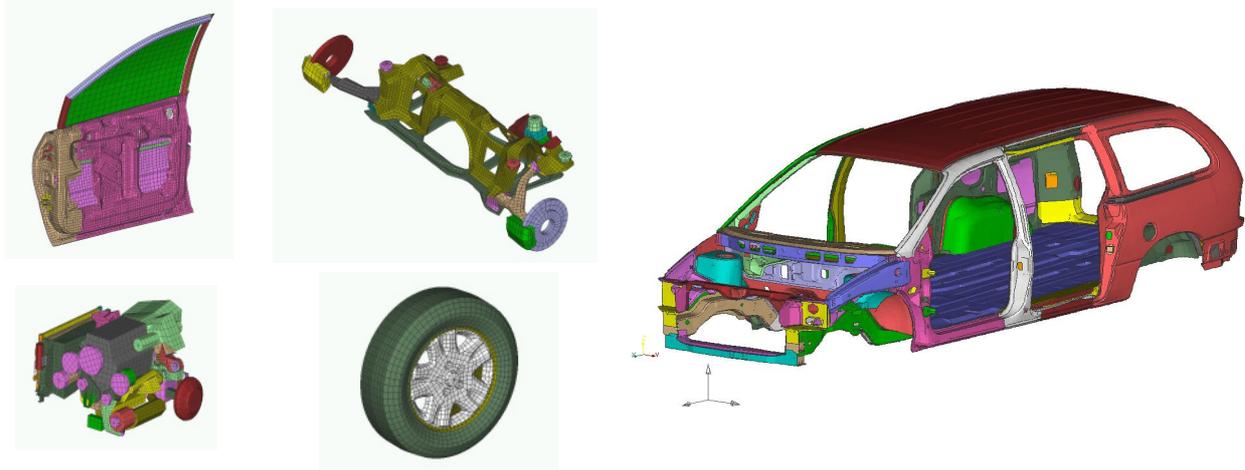


Bild 3: Fahrzeugkomponenten

Beispiele :

- Rohbau
- Tür
- Motor
- Vorderachse
- Rad

Um einen fehlerfreien Zusammenbau gewährleisten zu können, sind von den einzelnen Bauteilen folgende Bedingungen zu erfüllen:

1. Alle Verbindungen innerhalb des Bauteils sind definiert.
2. Geometrie des Bauteils ist über Elemente und Knoten komplett definiert.
3. Die Masse des Bauteils ist abgestimmt.
4. Alle Properties und Materialien sind definiert.
5. Mit einer geeigneten Lastfall-Beschreibung ist das Bauteil rechenbar.
6. Die Anbindepunkte zu Nachbarbauteilen sind gekennzeichnet.
7. Bauteilübergreifende Definitionen wie Gesamtkontakt , Gesamtschweißkontakt und die Objekte für Anfangsbedingungen sind gekennzeichnet.

## 2.2 Verbindungen

Die Verbindung der Bauteile erfolgt über Techniken, bei der das einzelne Bauteil seine eigenständige Rechenfähigkeit nicht verliert.

Nachfolgend werden die Möglichkeiten bei der Anbindung einer Tür an den Rohbau aufgezeigt.

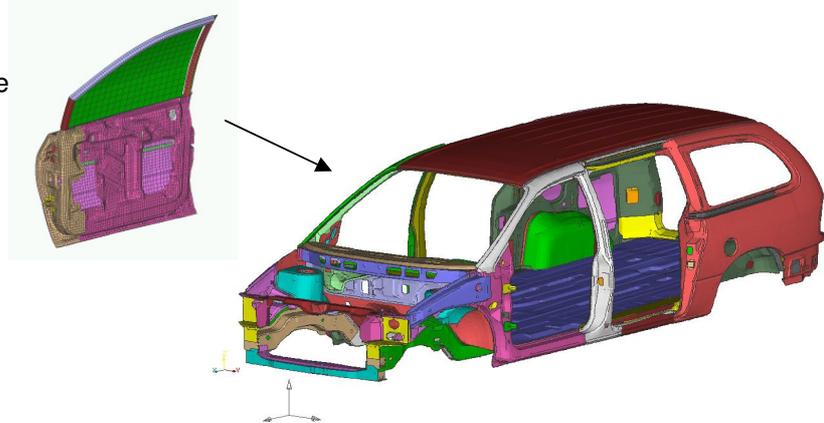


Bild 4: Verbindung

### 2.2.1 Verbindung über NRB-Elemente

In den Bauteilen gekennzeichnete Knoten werden bei koinzidenten Koordinaten mit einem NRB-Element verbunden.

```
*CONSTRAINED_NODAL_RIGID_BODY
```

### 2.2.2 Verbindung über Verschmelzen von Knoten

In den Bauteilen gekennzeichnete, koinzident Knoten an den Verbindungspositionen werden verschmolzen.

### 2.2.3 Schweiß-Verbindung

Gekennzeichnete Properties werden im Master- bzw. Slave-Set des Gesamtschweißkontakts ergänzt.

```
*CONTACT_TIED_SHELL_EDGE_TO_SURFACE
```

### 2.2.4 Klebe-Verbindung

Gekennzeichnete Properties werden in den Master- bzw. Slave-Sets der Klebekontakte ergänzt.

```
*CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET
```

### 2.2.5 Starrkörper-Verbindung

Gekennzeichnete Properties werden über eine „rigidbody merge card“ miteinander verbunden.

```
*CONSTRAINED_RIGID_BODIES
```

### 2.2.6 Starrkörper-Verbindung über extra Node

Gekennzeichnete Knoten werden über die „extra node card“ einem Starrkörper hinzugefügt.

```
*CONSTRAINED_EXTRA_NODES_NODE/SET
```

### 2.2.7 Ankleben von Knoten (Time-History-Knoten)

Gekennzeichnete Knoten werden über Klebekontakte an die Schalenstruktur geklebt.

\*CONTACT\_TIED\_NODE\_TO\_SURFACE

## 2.3 Bauteil übergreifende Definitionen

Aus der Interaktion der einzelnen Bauteile im Gesamtmodell ergeben sich zusätzliche Bauteilbedingungen bzw. werden bestehende Bedingungen modifiziert.

### 2.3.1 Kontakte

Die einzelnen Bauteile eines Zusammenbaus müssen mit allen anderen Bauteilen Kontaktverhältnisse eingehen können. Aus diesem Grund müssen alle Bauteile in einen Gesamtkontakt eingetragen werden.

Weiterhin wird für alle Schweißverbindungen des Typs \*CONTACT\_TIED\_SHELL\_EDGE\_TO\_SURFACE nach dem Zusammenbau ein Gesamtschweißkontakt gebildet.

Außerdem werden die Klebeverbindungen des Typs \*CONTACT\_TIED\_SHELL\_EDGE\_TO\_SURFACE\_Beam\_Offset und \*CONTACT\_TIED\_NODE\_TO\_SURFACE generiert.

### 2.3.2 Anfangsbedingungen

Einzelnen Bauteilen wird z. B. eine initial velocity zugewiesen.

### 2.3.3 Globaler Massenabgleich

Die Massen einzelner oder aller Bauteile werden nach dem Zusammenbau nochmals modifiziert, um die Gesamtmasse des Modells an die am realen Fahrzeug ermittelte Gesamtmasse anzupassen.

## 2.4 Control-Include

Das Control-Include enthält die Lastfalldefinitionen.

## 2.5 Koppel-Include

Das Koppel-Include enthält alle Objekte, die zur Verbindung der Bauteile erstellt werden. Dazu gehören:

- Verbindungen über NRB-Elemente
- Schweiß-Verbindungen
- Klebe-Verbindungen
- Starrkörper-Verbindungen

Hier werden auch alle Sets generiert die für bauteilübergreifende Definitionen (z. B. initial velocity) benötigt werden, sowie alle bauteilübergreifenden Definitionen selbst.

## 2.6 Master-File (Steuerdeck)

Das Master-File wird generiert und enthält die Aufrufe aller ausgewählten Bauteil-, Control- und Koppel-Includes.

## 2.7 Berechnungsvariante

Eine Berechnungsvariante setzt sich aus einem Satz von Files mit allen für den Lastfall benötigten Definitionen zusammen. Für diese kann z. B. während des Zusammenbaus ein eigenes Verzeichnis angelegt werden. Schematisch kann das folgendermaßen aussehen:

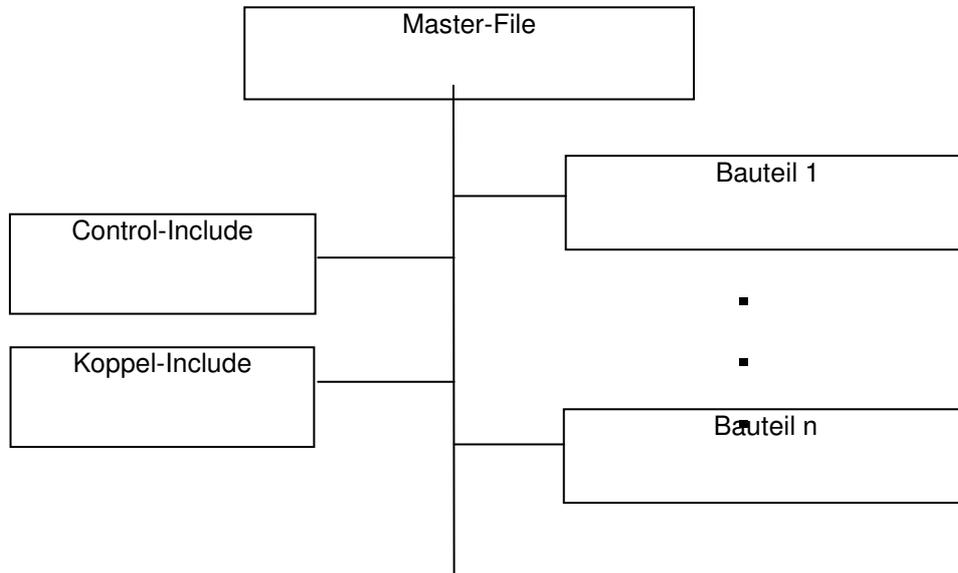


Bild 5: Include Schema

### 3 Funktionalitäten

#### 3.1 Prüfen einzelner Bauteile

Das Prüfen der einzelnen Bauteile erfolgt z. B. über eine im Vorfeld abgestimmte Vorgehensweise.

- Einführen von check Lastfällen
- Modalanalyse ( Rigid body modes )

#### 3.2 Der Assemblierungsprozess

Beim Assemblierungsprozess müssen für einen fehlerfreien Zusammenbau folgende Schritte durchgeführt werden:

- Aufbau des Master-Files aus der Liste aller selektierten Bauteile.
- Aufbau des Koppel-Include-Files. Generieren aller Verbindungen + bauteilübergreifenden Definitionen.
- Zusammenstellen aller für die Berechnungsvariante benötigten Files.

#### 3.3 Prüfen des Zusammenbaus

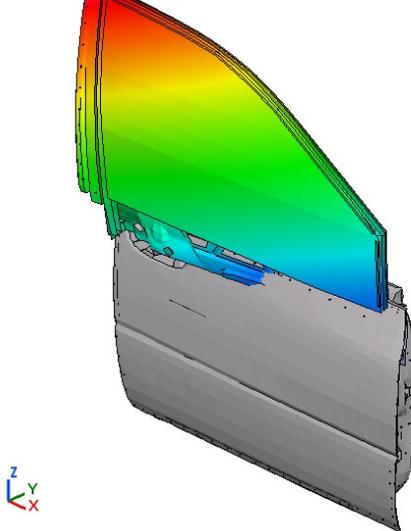
Da die Einzelbauteile in sich geprüft sind, kann die Prüfung des Zusammenbaus auf die Kontrolle der verwendeten Verbindungsmittel, definierten Anfangsbedingungen und die Verwendung der den Bauteilen zugewiesenen Nummernräumen beschränkt werden.

Das Modell wird auf offene Verbindungen untersucht. Außerdem wird die Vollständigkeit des Gesamtmodells geprüft, so dass das Fehlen einer Tür oder eines Rades dem Anwender angezeigt wird.

Die Prüfung auf offene Verbindungen kann z. B. über eine Modalanalyse erfolgen. Fehlerhafte oder unzureichende Verbindungen von Bauteilen können durch Kontrolle der Eigenformen gefunden werden.

Durchdringungen von Bauteilen im Gesamtkontakt müssen vermieden werden und können durch Testanrechnung durch folgende Meldung erkannt werden.

d3eigv : Displacements,Magnitude : STATE 1 ,TIME  
4.74817455E-01



Zusammenbau

d3eigv : Displacements,Magnitude : STATE 1 ,TIME  
4.74817455E-01

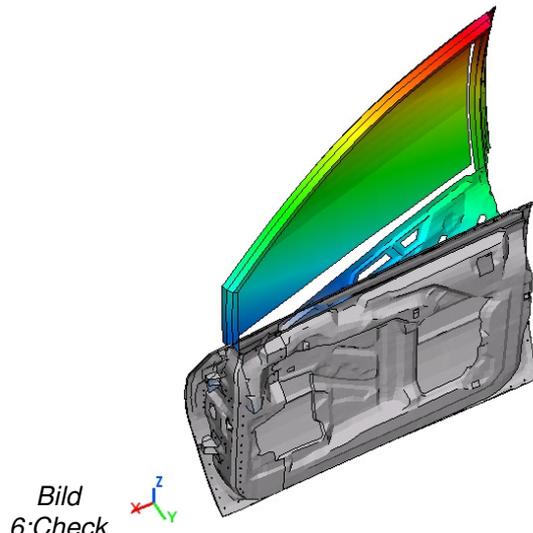


Bild  
6:Check

des

```
Node 241 has penetration on SI          2
  Current coords:  -0.141010E+04        -0.910000E+03        0.620900E+03
 Preferred coords: -0.140792E+04        -0.909358E+03        0.621968E+03
  difference:      0.217847E+01          0.642456E+00        0.106799E+01
```

Über alle Bauteile werden aufgebrachte Anfangsbedingungen, wie z. B. eine initial velocity, geprüft. Angezeigt werden dabei z. B. die Bauteile, für die noch keine Anfangsbedingungen definiert wurden. Werden von einzelnen Bauteilen die vorgegebenen Nummernräume nicht eingehalten, wird der Anwender darauf hingewiesen.

### **3.4 Bauteile verändern**

Um das Modell dem Entwicklungsprozess anzupassen werden für die Bauteile Veränderungen notwendig. Die Realisierung von Modifikationen kann dabei wie folgt passieren.

#### *3.4.1 Ohne Umgebung*

Das zu verändernde Bauteil wird einzeln betrachtet. Es ist dabei darauf zu achten, dass sich die speziell für die Anbindungen bearbeiteten Bereiche dabei nicht verändern, bzw. alle von den Modifikationen betroffenen Bauteile ebenfalls angepasst werden. Es ist somit notwendig, innerhalb eines Bauteils Bereiche zu schützen. Der Anwender wird auf mögliche Veränderungen dieser Bereiche über zusätzliche Abfragen hingewiesen.

#### *3.4.2 Mit Umgebung*

Das einfache Kombinieren von Bauteilen ermöglicht den schnellen Zusammenbau einzelner Bauteile. Auf diese Weise können bei der Modifikation eines Bauteils zu jeder Zeit die möglichen Auswirkungen auf die mitbetrachtete Umgebung erkannt werden. Das Vorgehen kann dabei wie folgt aussehen:

- Bauteil + Umgebung selektieren
- Assemblieren ohne Verbindung
- Einlesen in Preprozessor
- Bauteil ändern
- Auswirkungen auf die Umgebung erkennen
- Bauteil ausgeben

### **3.5 Selektion der gewünschten Komponenten**

Die Auswahl der einzelnen Bauteile kann aus einer Datenbank erfolgen. Diese kann die folgenden Eigenschaften besitzen.

- Komponenten darstellen mit Selektionsmöglichkeiten.
- Abhängigkeiten prüfen.

### **3.6 Verwaltung der Berechnungsvarianten**

Um bei einer Vielzahl von Versionen mit ebenso vielen Varianten den Überblick nicht zu verlieren, ist es sinnvoll, eine Verwaltung zu haben, mit der sich beides geordnet ablegen lässt.

## 4 Daten Ablage

Die Ablage von Bauteilen, die als Grundlage für Gesamtmodelle dienen kann auf vielfältige Weise erfolgen.

### 4.1 Daten im Keyword Format ablegen

Die Bauteile werden nach ihrer Überprüfung im rechenfähigen Keyword Format abgelegt.

### 4.2 Daten im Preprozessor Format ablegen

Die Bauteile werden nach ihrer Überprüfung im Modellformat des jeweiligen Preprozessors abgelegt.

#### 4.2.1 Medina bif

Bei der Ablage im Format Medina bif werden folgende Daten notwendig:

- bif
- control
- Verweis auf Datenbanken.

Für den Zusammenbau werden die einzelnen Bauteile über diese Daten mit Hilfe von MEDINA Interfaces (bifdyn) in ein DYNA3D-Deck umgewandelt.

#### 4.2.2 Andere Formate

Durch Integration von externen Schnittstellen können die Module grundsätzlich auch in „beliebigen“ weiteren Formaten abgelegt und verwaltet werden.

### 4.3 Gemischte Datenablage

Das Ablegen von Daten, die als Modellgrundlage dienen, in unterschiedlichen Formaten stellt hohe Anforderungen an die Tools für den Zusammenbau, da alle bestehenden Modellinformationen und Einstellungen der einzelnen Bauteile ohne Verluste in das Format des Zusammenbaus übertragbar sein müssen.

## 5 Zusammenfassung

Mit MEDINA/SDM wird der Grundstein gelegt für eine übersichtliche Verwaltung von komplexen Modellstrukturen auf Basis von Modulen.

Zusätzlich wird hier eine Arbeitsweise etabliert welche den Einsatz von größeren Arbeitsgruppen in der Simulation erlaubt, innerhalb derer unterschiedliche Ausprägungen der Spezialisierung der Mitarbeiter gegeben ist.