

Ralph Stelzer · Karl-Heinrich Grote · Klaus Brökel  
Frank Rieg · Jörg Feldhusen (Hrsg.)

# **ENTWERFEN ENTWICKELN ERLEBEN**

Methoden und Werkzeuge in der Produktentwicklung



**10. Gemeinsames Kolloquium Konstruktionstechnik  
KT2012 | Residenzschloss Dresden | 14.–15. Juni 2012**



Stelzer · Grote · Brökel · Rieg · Feldhusen (Hrsg.)

**ENTWERFEN ENTWICKELN ERLEBEN**

Methoden und Werkzeuge in der Produktentwicklung

10. Gemeinsames Kolloquium Konstruktionstechnik KT2012



Ralph Stelzer · Karl-Heinrich Grote · Klaus Brökel  
Frank Rieg · Jörg Feldhusen (Hrsg.)

# **ENTWERFEN ENTWICKELN ERLEBEN**

Methoden und Werkzeuge in der Produktentwicklung

Entwickeln – Entwerfen – Erleben.  
Methoden und Werkzeuge in der Produktentwicklung  
10. Gemeinsames Kolloquium Konstruktionstechnik (KT2012)

Herausgeber:

Prof. Dr. Ralph Stelzer (Technische Universität Dresden)  
Prof. Dr. Karl-Heinrich Grote (Otto-von-Guericke-Universität Magdeburg)  
Prof. Dr. Klaus Brökel (Universität Rostock)  
Prof. Dr. Frank Rieg (Universität Bayreuth)  
Prof. Dr. Jörg Feldhusen (RWTH Aachen)

Wir bedanken uns für die Unterstützung bei  
ma design, Tedata, Continental, xPLM, B.I.M. Consulting und Reiss Büromöbel

**ma design**  
//ENGINEERING

**Continental** 

**B.I.M.**  
consulting

**TEDATA**

**xPLM**  
Solution

**REISS**

Bibliografische Information der Deutschen Nationalbibliothek  
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der  
Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind  
im Internet über <http://dnb.d-nb.de> abrufbar.

Bibliographic information published by the Deutsche Nationalbibliothek  
The Deutsche Nationalbibliothek lists this publication in the Deutsche  
Nationalbibliografie; detailed bibliographic data are available in the  
Internet at <http://dnb.d-nb.de>.

ISBN 987-3-942710-80-0

© 2012 TUDpress  
Verlag der Wissenschaften GmbH  
Bergstr. 70 | D-01069 Dresden  
Tel.: 0351/47 96 97 20 | Fax: 0351/47 96 08 19  
<http://www.tudpress.de>

Alle Rechte vorbehalten. All rights reserved.  
Layout und Satz: Sandra Olbrich/Technische Universität Dresden.  
Umschlaggestaltung: TU Dresden, Illustration Audi A6 Limousine © 2012 Audi AG

Bettina Alber-Laukant, Markus Zimmermann,  
Florian Nützel, Michael Frisch & Frank Rieg

# **Anforderungen an die grafische Oberfläche eines FE-Systems aus Sicht des Ingenieurs**

## **Einleitung**

In der Produktentwicklung hat sich die numerische Simulation mittels Finite Elemente Analyse als Standardverfahren etabliert. Die Wandlung von einem Spezialistenwerkzeug zu einem generellen Ingenieurtool bringt mit sich, dass theoretisches Wissen zur FEA beim Benutzer nicht vorausgesetzt werden kann und demzufolge im Programm geeignet implementiert sein muss. Ingenieure sind im Umgang mit CAE-Tools (FEM, CFD, CAD) und theoretischen Methoden (VDI 2221, DSM, DfX) im Produktentstehungsprozess vertraut. In diesem Zusammenhang ist die aktuelle Herausforderung eine Verbindung beider etablierten Werkzeugklassen zu schaffen, die bisher nicht erfolgt ist. Es fehlt die benutzerspezifische Anpassung von computergestützten Werkzeugen an die Denkweise von Ingenieuren und deren Nutzungskontext. Methodische Wissensbasen, Retrieval-Möglichkeiten und Suchroutinen für abgeschlossene Projekte sind in CAE-Programmen nicht integriert. Ergänzt werden diese Nachteile durch die hohen Anschaffungskosten von Software in Kombination mit schlechter Usability.

Das Ziel des vorliegenden Beitrags ist die Entwicklung einer Strategie für die Integration methodischer Unterstützung in eine Simulationsumgebung. Hierzu muss die Software zunächst essentielle Qualitätsmerkmale aufweisen, um für die FEA ähnliche Qualitätsstandards zu erreichen, wie sie in anderen Schritten des Produktentwicklungsprozesses üblich sind. Weiterhin muss eine Prozessunterstützung erfolgen, da für eine erfolgreiche FE-Berechnung viele Entscheidungsschritte nötig sind. Aus Akzeptanzgründen ist die Nutzbarkeit durch eine intuitive Benutzerführung, wie die Verwendung von Ablaufschemata und Best Practices in den Hilfedokumenten und den Aufbau der Menüs nach Workflowgesichtspunkten obligatorisch. Durch die umfassende Unterstützung des Anwenders kann die FEA sowohl Experten als auch Anfängern im jeweils erforderlichen Umfang bereitgestellt werden. Dies verbessert die Rahmenbedingungen für die Verwendung computergestützter Werkzeuge innerhalb des Produktentwicklungsprozesses.

### **Zielsetzung**

Die Verwendung von computergestützten Werkzeugen für die Produktentwicklung können bei effizientem Einsatz entscheidende Vorteile im Wettbewerb erzielen. Doch vor allem ein schlechtes Pre- und Postprocessing der zur Verfügung gestellten Informationen verursacht die meisten Verzögerungen im Bereich Engineering Design. Auch fehlende Simulationsstrategien und Handlungsempfehlungen senken die Effizienz im Entwicklungsprozess. Dabei würde die Vermeidung unnötig durchgeführter Schritte im FE-Workflow Kosten stark senken. Das Primärziel ist daher die Entwicklung einer Plattform für die Integration einer unterstützenden Engineering-Methode in die Simulationsumgebung, um die ständig steigende Komplexität der FEA besser und effizienter handhaben zu können.

### **Anforderungen an aktuelle FEA-Systeme**

Um die Anforderungen des Benutzers an moderne FEA-Systeme effektiv in der Benutzeroberfläche umzusetzen, müssen diese zunächst definiert werden. Dazu können Benutzerforen, Workshops, Support-E-mails oder Interviews mit Nutzern von FEA-Software ausgewertet

werden. In einem ersten Schritt werden das unterschiedliche fachspezifische Wissen und die verschiedenen Problemfelder erfasst. Im nächsten Schritt erfolgt eine Untersuchung der bestehenden Benutzeroberflächen und Hilfe-Systeme in Softwaresystemen, um den Stand der Technik zu analysieren sowie deren Vor- und Nachteile bewerten zu können. Mit Hilfe dieses Vorgehens konnten folgende Problembereiche aktueller FEA-Software heraus gearbeitet werden:

- Eine mangelnde Verknüpfung zwischen gemeinsamen verwendeten Software-Tools verursacht Schnittstellenprobleme für den Datenaustausch
- Die Berücksichtigung unterschiedlichen Knowhows der Benutzer findet nicht statt
- Die Programme entsprechen nicht den vom Ingenieur erwarteten Qualitätsmerkmalen
- Es fehlen Best-Practices zur Unterstützung der Anwender, ohne dabei die eigenen Kreativität einzuschränken
- Eine situationsabhängige Benutzer-Unterstützung ist nicht vorhanden
- Bisher wird eine abschreckende Wirkung durch die Informationsflut vieler Hilfe-Dokumente erzielt, die Hilfe-Artefakte sind auf Dokumente beschränkt
- Keine Möglichkeiten abgeschlossene Projekte und Suchroutinen für Analogien aufrufen zu können
- Definition von Qualitätskriterien und programmtechnische Umsetzung

Um sich dieser Herausforderung, der Entwicklung einer neuen graphischen Oberfläche zu stellen, musste die gesamte Idee des User-Supports innerhalb der Simulation grundlegend neu überdacht werden. Das Thema Workflow wird dazu aus verschiedenen Blickwinkeln betrachtet. Die konstruktive Bauteilauslegung mittels FEA stützt sich auf festgelegte Schritte, welche für eine sichere und zuverlässige Auslegung notwendig sind. Das primäre Ziel ist, eine Software zu entwickeln, welche den Qualitätsstandards von Ingenieuren gerecht wird und dadurch Akzeptanz findet. In der Literatur finden sich diverse Ansätze von Qualitätsmerkmalen für Softwaresysteme, wobei der Begriff »Qualität« subjektiv und in der praktischen Anwendung schwierig zu definieren ist. Legt man die inter-

ationale Norm ISO/IEC 9126 zugrunde, findet eine Definition von interner und externer Qualität der Software statt, wobei der Softwareentwickler typischerweise die Merkmale der inneren Qualität, wie Portierbarkeit, der Softwareanwender die äußeren Merkmale, wie Benutzbarkeit im Blick hat.

Im Kontext des Ingenieurs sind natürlich die Richtigkeit der Ergebnisse und die Reproduzierbarkeit eines Ergebnisses unter gleichen Randbedingungen essentiell. Die Software trägt durch korrekte Programmerroutinen dazu bei, das gewünschte Ergebnis zu erhalten, gleichzeitig kann jedoch die Benutzerunterstützung in der FEA an dem Punkt ansetzen, dass durch korrekte Wahl des FE-Typs, der Netzparameter und Randbedingungsangabe ein verlässliches Ergebnis entsteht. Die Hauptaufgabe der Benutzeroberfläche ist die Benutzerfreundlichkeit. Die Bedienungsreihenfolge stringent aufzubauen, den Ablauf konsequent von links nach rechts, von oben nach unten vorzugeben, lässt ein routinemäßiges Vorgehen bei mehrmaliger Anwendung der Software zu. Unterstützend finden Assoziationen, Analogien und redundante Signale bei der Verwendung der Icons in der graphischen Darstellung Verwendung, um damit die Aussagekraft und Wiedererkennungsfähigkeit zu erhöhen.

Die Programmierung trägt hauptsächlich zur Effizienz bei. Durch die Verwendung einer Programmiersprache, wie z.B. C, wird die Geschwindigkeit des Programmes erhöht, denn im Allgemeinen ist die Schnittstelle zu allen Betriebssystemen eine C-Schnittstelle, so dass C-Programme den schnellsten ausführbaren Maschinencode liefern. Die Benutzerunterstützung kann aber durch Zugriff auf die Hardware die Nutzung der vorhandenen Rechnerkapazitäten verbessern, indem bei der Berechnung eine Verwendung eines mehrprozessorfähigen Solvers auf mehrere CPUs erfolgt. Ein weiterer Vorteil bei der Verwendung einer international genormten Programmiersprache ist die Portabilität des Programmes auf unterschiedlichste Betriebssysteme, da sich neben Windows-Bürorechnern beispielsweise zunehmend MAC Betriebssysteme durchsetzen.

Die Änderbarkeit der Software ist im Bereich der Ingenieursanwendungen sehr wichtig, nachdem zunehmend gekoppelte Simulatio-

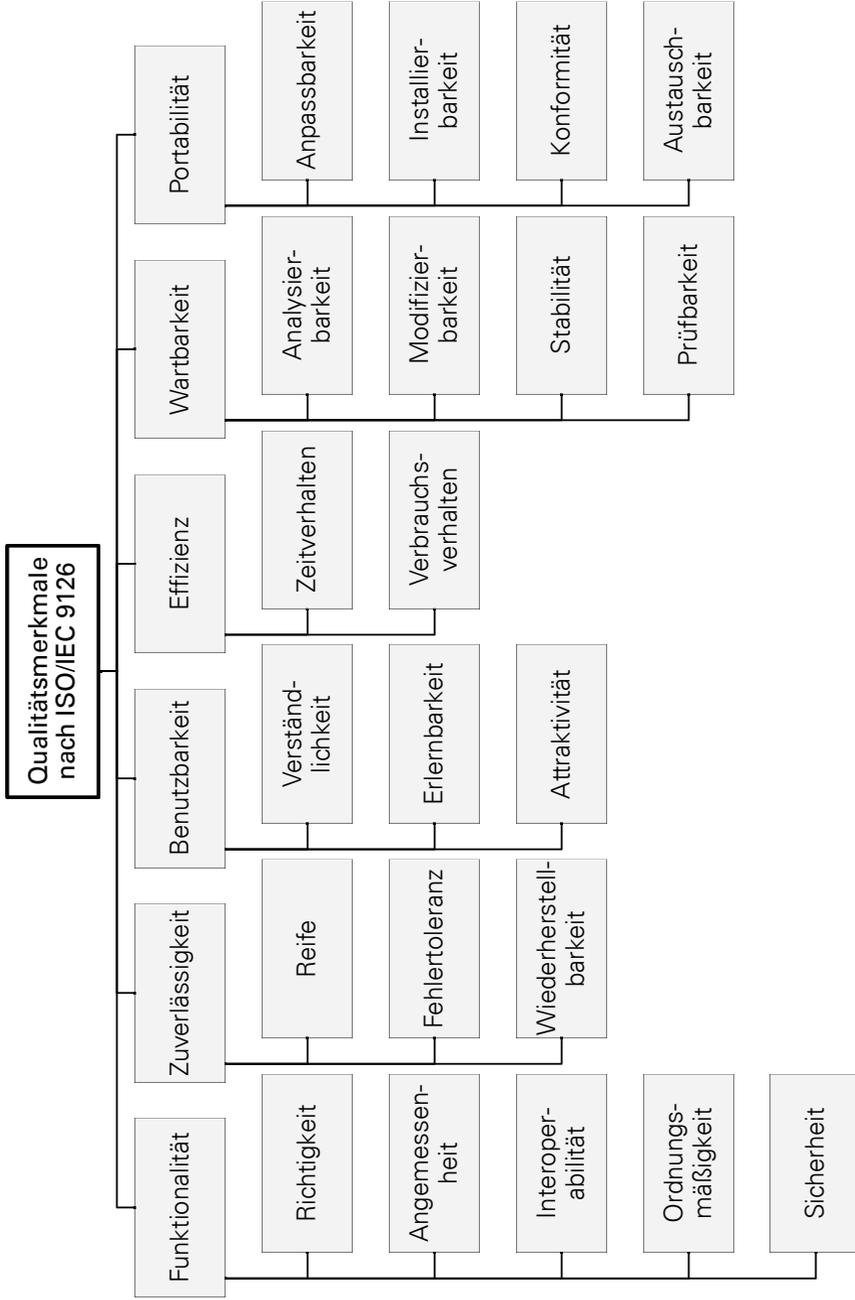


Abbildung 1: Qualitätsmerkmale von Softwaresystemen nach ISO/IEC 9126, Teil 1 (ISO/IEC 2001)

|                    |   |   |   |
|--------------------|---|---|---|
| Redundant Signale  | <br>Contour plot | <br>Solver                   | <br>Licht         |
| Assoziationsketten | <br>Superelement | <br>Knotenpicking            | <br>Einstellungen |
| Analogien          | <br>Knoten       | <br>Thermische<br>Berechnung | <br>Tastatur      |

Abbildung 2: Aussagekraft und Wiedererkennungswert von Icons

nen Verwendung finden. Dies ist programmiertechnisch durch einen modularen Aufbau der ProgrammROUTINEN möglich. Seitens der Benutzeroberfläche kann durch wechselnde Seitenmenüs und daraus aufrufbare Pop-Up Fenster die Variabilität gewährleistet werden.

Für die Erlernbarkeit des Programmes stehen Ingenieuren mehrere methodische Vorgehen zur Verfügung, die geeignet in das Programm eingebunden werden müssen. Üblicherweise existieren zwei Möglichkeiten den Benutzer bei seiner Aufgabenstellung zu unterstützen: Auf der einen Seite gibt es ein umfangreiches Handbuch, mit allen verfügbaren Informationen zu einem Programm oder einem Prozess, auf der anderen Seite gibt es immer ausgefeiltere Systeme, die mit Hilfe einer Checkliste, wie sie in klassischen Workflow-Management-Systemen verwendet werden, anzeigen, welche Schritte bereits ausgeführt wurden bzw. was noch zu tun sind. Beide Systeme eignen sich nur für mehr oder weniger fortgeschrittene Anwender. Der Grund dafür liegt in der Tatsache, dass zu keinem Zeitpunkt des Prozessschrittes eine genaue Angabe der benötigten Informationen gemacht wird (Zapf et al. 2010).

Die Entwicklung eines Prozessführungstools, hier SPIDER (systematic procedures for simulation driven engineering research) genannt, stellt ein Konzept der situationsabhängige Benutzerunterstützung dar, welches verschiedene methodische Ansätze sinnvoll kombiniert. Neben einer einfachen Führung durch den Prozess, muss der Anwender mit den für die Lösung seiner Problemstellung notwendigen Informationen versorgt werden. Durch die Kombination von kontextspezifischen Tooltips, nützlichen und informativen Dokumenten, Best-Practices, Videos, Entscheidungsmatrizen (DSM), Benutzerabfragen sowie durch den internen Abruf und die Auswertung systemspezifischer Daten (Prozessorleistung, Graphikkarte, etc. ist es möglich, den Benutzer mit den Informationen zu versorgen, die er aktuell benötigt ohne ihn dabei mit überflüssigen Angaben zu überladen. Das Verfahren selbst wird an Hand eines praktischen Beispiels, des FEA Systems Z88Aurora, das vom Lehrstuhl für Konstruktionslehre und CAD der Universität Bayreuth entwickelte wurde, erklärt. Durch ein zusätzlich aktiviertes SPIDER-Andockfenster, welches den Workflow der jeweiligen Simulation enthält, können schnell verschiedene Informationsartefakte kontextbezogen abgerufen werden. Dabei wird der Fortschritt der Simulation ebenso wie die Ergebnisse dokumentiert.

### **Workflowdefinition**

Um für einen Workflow die verschiedenen Abhängigkeiten zwischen den einzelnen Prozessschritten zu definieren, müssen diese natürlich bekannt sein. Diese einzelnen Schritte können durch Sub-Prozesse beschrieben oder mit Hilfe von Design-Structure-Matrizen erfasst werden (Nützel et al. 2011). Damit können standardisierte Verfahren für die CAx-Prozesskette verwendet werden. Zum Einsatz kommt hier dafür das Prinzip ICROS (intelligent cross-linked simulation), das Prozesse chronologisch und sachlich logisch verknüpft (Alber et al. 2007).

Der Aufbau eines Berechnungsmodells unterteilt sich in unterschiedliche Schritte, die nacheinander abgearbeitet werden müssen, bis die Berechnung gestartet werden kann: Bereitstellung der Bauteilstruktur (Import) – Vorbereitung (Preprocessing) – Berech-

nung (Solver) – Analyse und Interpretation der Ergebnisse (Postprocessing): Das sind üblicherweise die vier großen Abschnitte einer FEA (Abbildung 3).

Der erste Schritt ist der Datei-Import. Die Auswahl eines geeigneten Austausch-Dateiformats ist immer noch eine kritische und wichtige Fragestellung. Einige Formate (STEP, STL, etc.) sind weit verbreitet, was nicht bedeutet, dass alle implementierten Datenstrukturen ohne Einschränkungen übertragen werden. Diese gebräuchlichsten Schnittstellen müssen in der Software enthalten sein, um eine breite Palette praktischer Anwendungsfälle abzudecken (Troll 2009).

Um Bauteile berechnen zu können, müssen sie durch ein FE-Netz diskretisiert werden. Erst dann können die Randbedingungen und Lasten aufgebracht werden. Normalerweise enthalten FE-Programme nur ein Vernetzungsmodul. Z88Aurora stellt verschiedenen Free- und Mapped-Mesher mit diversen einstellbaren Vernetzungsparametern bereit. Damit kann in Abhängigkeit der jeweiligen Aufgabenstellungen die FE-Netzqualität angepasst werden.

Um realistische Lastsituationen simulieren zu können, müssen die entsprechenden Randbedingungen und Lasten schnell und sicher von Benutzer auf das vernetzte Bauteil aufgebracht werden.

Die vorbereitete FE-Aufgabenstellung wird anschließend mit Hilfe eines FE-Solvers gelöst. Z88Aurora stellt in Abhängigkeit des FE-Modells und der verwendeten Computer Hardware verschiedene FE-Solver bereit. Damit kann das FE-Modell bzgl. von Kriterien wie Berechnungszeit und Speicherbedarf effektiv bewertet werden.

Das Postprocessing zeigt die Ergebnisse wie Verschiebungen, Spannungen, Kräfte und andere Informationen an, um die Bauteile nach den Auslegungskriterien bewerten zu können.

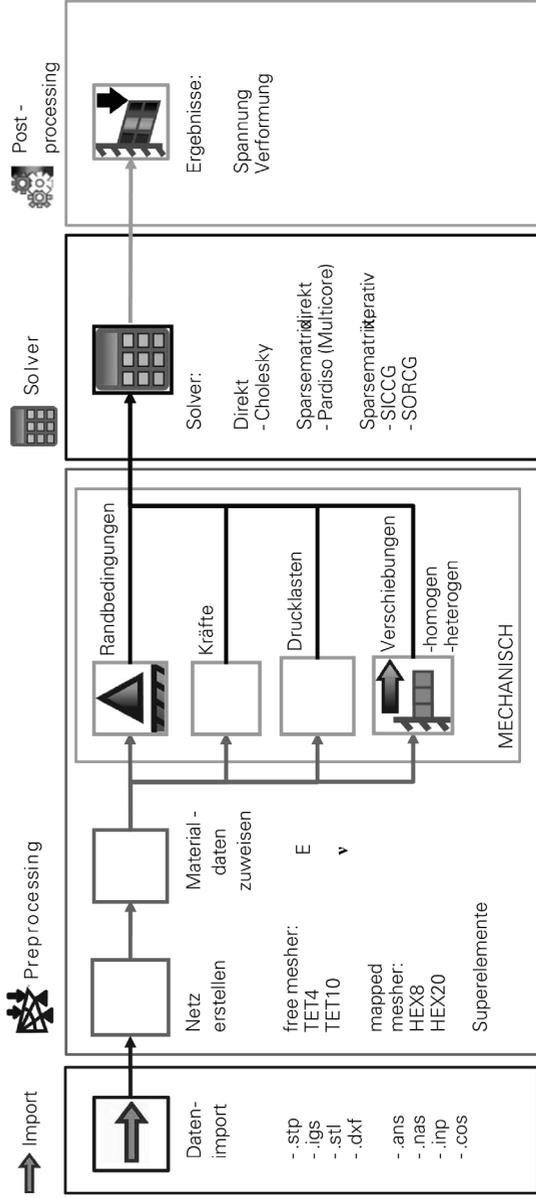


Abbildung 3: Workflow für eine statisch-mechanische FEA

## SPIDER

Die kontextsensitive FE-Wissensbasis SPIDER wurde sowohl zur effektiven Unterstützung von Anfängern als auch für Experten auf dem Gebiet der FEA konzipiert. Effektiv bedeutet in diesem Zusammenhang, dass das angestrebte Ergebnis mit höherer Wahrscheinlichkeit oder mit geringerem Risiko mit einer Kombination von Techniken, Methoden und Werkzeugen erreicht wird, als mit einer anderen Vorgehensweise.

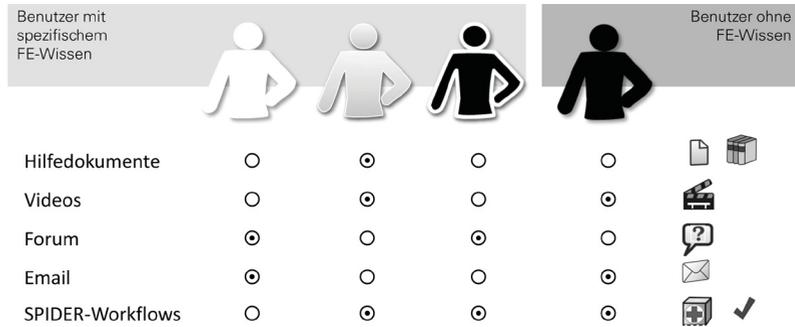


Abbildung 4: Einteilung der FEA-Anwender

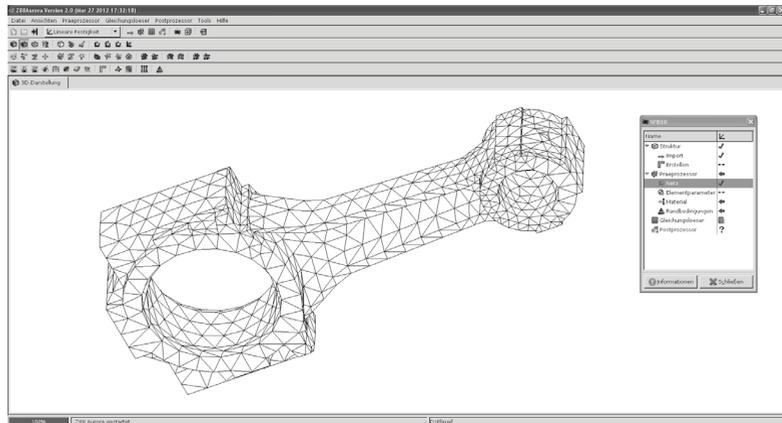


Abbildung 5: Integration von SPIDER innerhalb des FE-System Z88Aurora

Jeder einzelne Schritt des aktuellen FE-Prozesses wird mit einer Reihe von Hilfe-Funktionen und –Dokumenten unterstützt (Abbildung 4), die je nach Benutzerwissen ausgewählt werden können. Eine Checkliste zeigt den Fortschritt im FE-Workflow.

Die Benutzer des FEA-Systems können nach ihrem FE-Wissen in zwei Gruppen eingeteilt werden: In die Benutzergruppe mit bzw. ohne FE-Hintergrundwissen. Z88Aurora stellt mehrere Hilfe-Funktionalitäten (Handbücher, Mail-Support, User-Forum) bereit, die dem gängigen Benutzerverhalten angepasst wurden (Abbildung 4):

- Fortgeschrittene Benutzer der ersten Gruppe (links außen) benötigen nur wenig Hilfe, vor allem für sehr spezielle Fragen, die per Email oder im Hilfe-Forum beantwortet werden
- Weniger FE-erfahrene Ingenieure lesen die Anleitungen, um die Grundlagen zu verstehen oder verwenden die Videos
- FE-Benutzer mit geringem FE-Wissen, die selten FEA durchführen und Handbücher nicht lesen, werden durch die Video-Tutorials und Support-Emails problem-spezifisch unterstützt

Wird die SPIDER-Unterstützung in Z88Aurora gestartet, kann mit Hilfe einer Checkliste der Fortschritt im FE-Workflow angezeigt werden (Abbildung 5). Grüne Haken deuten an, dass ein Prozessschritt vollständig und richtig durchgeführt wurde und rote Kreuze, dass für die Durchführung der FEA noch weitere Angaben benötigt werden. So kann bspw. der Solver erst gestartet werden, wenn alle Angaben des Preprocessings vollständig absolviert wurden. Zusätzlich kann die noch durchzuführende Modellierungsdauer abgeschätzt werden. Klickt der Benutzer auf das Prozess-Icon, gelangt er direkt in das entsprechende Menü. Zudem besteht die Möglichkeit alle zum Prozessschritt notwendigen Dokumente und Videos direkt anzusehen.

## Durchführung

Die praktische Nutzung von SPIDER wird anhand der Implementierung in das FEA-System Z88Aurora gezeigt. Natürlich könnte diese Methode zur Benutzerunterstützung auch in ein anderes Software-System integriert werden. Die Anforderungen an ein FEA-Programm, das mit einem Prozessunterstützungssystem ausgestattet werden soll, sind:

- Eine anpassbare Benutzeroberfläche
- Verschiedene problemspezifisch modifizierbare Solver
- Modulares Programmkonzept anpassbar an verschiedene Benutzer mit unterschiedlichen FE-Wissen
- Abfrage situationspezifischer Daten zur Benutzerunterstützung

In fast der gleichen Weise muss das Programm einen Zugang zu den Programmroutinen bieten, muss demnach funktional als auch erweiterbar sein. Die Entwicklung neuer Funktionalitäten, insbesondere die Integration neuer Module in einer funktionalen Oberfläche,

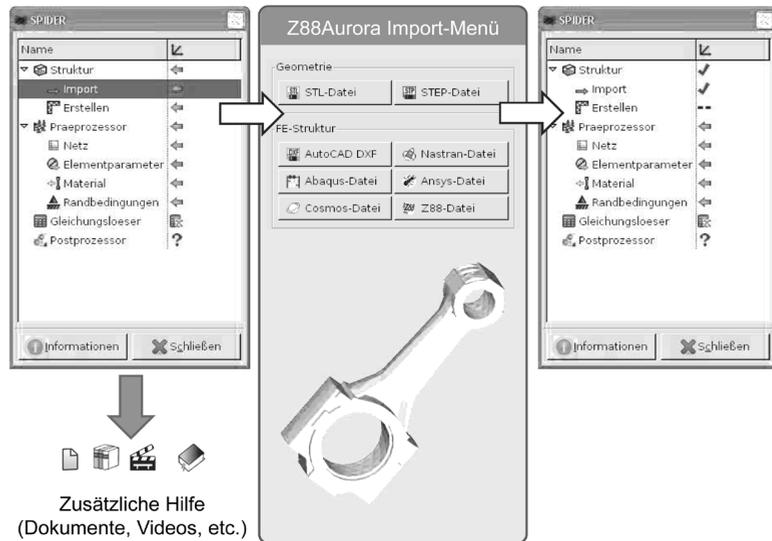


Abbildung 6: Datei-Import mit Benutzerunterstützung im FE-System Z88Aurora

muss einfach zu handhaben sein. Zudem soll die Integration neuer FE-Solver unkompliziert sein. Um das Programm schnell zu halten, erfolgt eine Aufteilung in einzelne Module: eine grafische Benutzungsoberfläche, die FE-Solver und Ergänzungsmodule.

Startet der Benutzer die SPIDER-Unterstützung öffnet sich ein Menü, indem von oben nach unten der Fortschritt im FE-Workflow angezeigt wird. Der Benutzer ist angehalten die einzelnen Menüpunkte in dieser Reihenfolge durchzuarbeiten, um effizient zu handeln. Zunächst wird der Benutzer aufgefordert eine Struktur zu importieren. Klickt er im SPIDER-Menü auf Import, öffnet sich das Import-Menü von Z88Aurora. Dort besteht die Möglichkeiten diverse Geometrie- bzw. FE-Struktur-Formate zum Import auszuwählen. Unerfahrene FE-Benutzer können die dort angebotenen Hilfedokumente oder Video-Tutorials nutzen, falls sie nicht wissen, welches Dateiformat sie z.B. als Export-Format im CAD-System wählen sollen oder welche Export-Parameter für das gewählte Export-Format im CAD-System zu wählen sind (Abbildung 6). Ist der Datei-Import erfolgreich abgeschlossen, erscheint im SPIDER-Menü ein grüner Haken. Analog erfolgt die Benutzerunterstützung für z.B. Vernetzung, Randbedingungen, Solver, Ergebnisdarstellung, etc.

Ein weiterer Anwendungsbereich für SPIDER ist die Verwendung und Handhabung der Picking-Funktion in Z88Aurora (Abbildung 7) mit der Randbedingungen und Materialzuweisungen vorgenommen werden. Mit Hilfe von z.B. Video-Tutorials wird die Selektion einzelner Bauteilbereiche erklärt. Nach kurzer Einarbeitung wird der Benutzer jedoch bemerken, wie intuitiv das Picking funktioniert, da auf die Markierungsfunktion vertrauter Office-Programmen zurückgegriffen wird. Das Picking-Menü bietet auch die Möglichkeiten verschieden ausgewählte Bauteilsektionen mit Hilfe von Booleschen Operationen zu bearbeiten.

Eine weitere unterstützende Funktionalität von SPIDER ist die Einbindung der FE-Modellspezifikationen in Kombination mit der vorhandenen Hardware zur Auswahl des FE-Solvers. Abhängig von der jeweiligen Situation wird der am Besten geeignete FE-Solver vorgeschlagen (Tabelle 1).

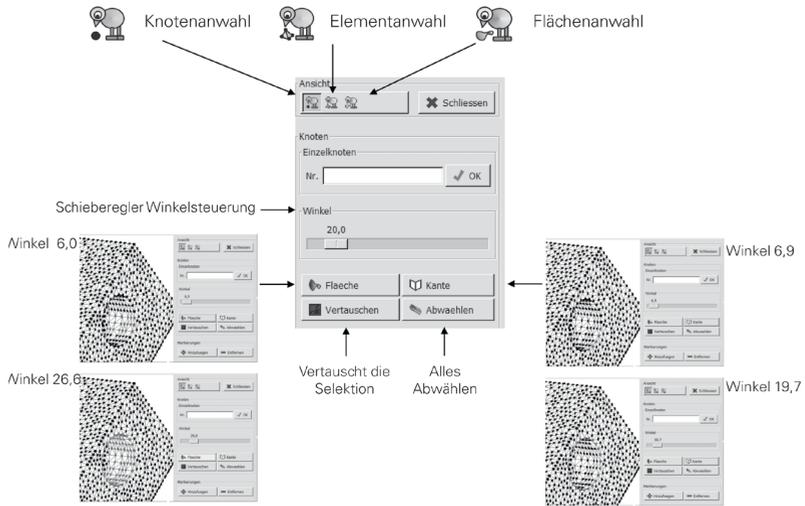


Abbildung 7: Picking-Funktion innerhalb des FE-System Z88Aurora

| FE-Solver                     | Cholesky              | SOR-CG  | SIC-CG  | Pardiso                     |
|-------------------------------|-----------------------|---|---|-----------------------------|
| <b>Typ Lösungsverfahren</b>   | direkt                | iterativ  | iterativ  | Direkt                      |
| <b>Speicherformat</b>         | Jennings              | sparse  | sparse  | Sparse                      |
| <b>Mehrprozessorfähigkeit</b> | nein                  | nein  | nein  | Ja                          |
| <b>Speicheranforderung</b>    | sehr groß             | minimal   | minimal   | Groß                        |
| <b>FE-Modellgröße</b>         | klein                 | sehr groß   | sehr groß   | je nach RAM                 |
| <b>FE-Problemstellung</b>     | lange Berechnungszeit | Ergebnisse sind von Konvergenz des Gleichungssystems abhängig | Ergebnisse sind von Konvergenz des Gleichungssystems abhängig | mehrere CPU-Kerne notwendig |

Tabelle 1: Situationsabhängiger FE-Solver-Vorschlag

## Qualität, Ergebnisse und Ausblick

Rechnergestützte Werkzeuge haben die Produktentwicklung in eine virtuelle Umgebung transferiert. Viele CAx-Programme sind nicht mit dem herkömmlichen methodischen Vorgehen eines Konstrukteurs verknüpft. Die verschiedenen Software-Hersteller bieten dem Benutzer häufig keine problemspezifische Hilfestellung in Echtzeit. Daraus resultieren mehrere Probleme:

- Aufgrund der schnellen Software-Entwicklung entstehen Divergenzen zwischen Programm-Funktionalität und Benutzer-Training
- Realistische Simulationen erfordern einen hohen Schulungsaufwand
- Änderungen zwischen verschiedenen Software-Versionen verursachen Probleme bei der Wiederverwendung von existierenden Datenpools
- Programmpotentiale werden wegen fehlender Vorgehensweisen meist nicht vollständig ausgeschöpft
- Fehlen eines richtigen User-Supports und einer Best Practices-Vorgehensweise in übersichtlicher Art u. Weise
- Datenbanksysteme können das Aufrufen und Bearbeiten alter Datensätze erschweren
- Verwendung von numerischen Methoden impliziert eine kritische Prüfung der Ergebnisse
- Ergebnisinterpretation ist von entscheidender Bedeutung für die Simulationen
- Gefahr der Fehlinterpretation

Das Prozessführungstool SPIDER verknüpft am Beispiel des FEA-Systems Z88Aurora mit der Entwicklungsmethodik und liefert einen Ansatz, um solche Probleme im Produktentstehungsprozess zu vermeiden. Mit der Hilfe von SPIDER erfolgt eine benutzerspezifische Anpassung von computergestützten Werkzeugen an die Denkweise von Ingenieuren und deren Nutzungskontext. Durch das vorgestellte Verfahren soll der Produktentwicklungsprozess effizienter gestaltet werden. Die Methodik hinter SPIDER soll im Grundsatz für diverse andere Software-Systeme anwendbar sein und sich nicht ausschließlich auf die FEA beschränken.

## Literaturverzeichnis

- Alber, B.; Rieg, F.; Hackenschmidt, R. 2007: Bauteilentwicklung mit Hochleistungskunststoffen.  
In: MP Materials Testing, Vol. 49, p. 402-408, 2007.
- Troll, A. 2009: Systematische Einbindung der CAE-Werkzeuge in den Entwicklungsprozess. In:  
Meerkamm, H.; Henrich, A.; Jablonski, S.; Krcmar, H.; Lindemann, U.; Rieg, F.: Flexible  
Prozessunterstützung in der Produktentwicklung: Prozesse – Daten – Navigation.  
Shaker-Verlag GmbH, Aachen, 2009.
- Zapf, J.; Alber-Laukant, B.; Rieg, F. 2010: Customized Design Processes of Polymer Parts by  
Computer Aided Tools. In: Proceedings of the Design 2010, Dubrovnik, 2010.
- Nützel, F.; Alber-Laukant, B.; Neidnicht, M.; Wehmann, C.; Rieg, F. 2011: Die Korrelation von  
Modell- und Modellierungsparametern in der Kontaktanalyse bei Maschinenelementen.  
In: Veröffentlichung zu 9. Gemeinsames Kolloquium Konstruktionstechnik 2011, Rostock.
- ISO/IEC 9126-1: Software Engineering Product Quality Part 1: Quality Model, 2011.

## Kontakt

Dr.-Ing. Bettina Alber-Laukant  
Dipl.-Ing. Markus Zimmermann  
Dipl.-Ing. Florian Nützel  
Michael Frisch, M.Sc.  
Lehrstuhl für Konstruktionslehre und CAD  
Universität Bayreuth  
Universitätsstraße 30  
95447 Bayreuth