

Vorbereitung von CAD-Konstruktionsdaten für den RP-Einsatz – eine Schnittstellenproblematik

Dipl.-Ing. Stéphane Danjou *Universität Duisburg-Essen Institut für Produkt Engineering / IPE Produktionstechnologie und Produktentwicklung*

Prof. Dr.-Ing. Dipl.-Math. Horst Koehler *Universität Duisburg-Essen Institut für Produkt Engineering / IPE Produktionstechnologie und Produktentwicklung*

Zusammenfassung

Während zunehmend Verfahrensverbesserungen der RP-Technologie Gegenstand aktueller Entwicklungen sind, ist die datentechnische Integration des RP-Prozesses in die CAx-Landschaft nach wie vor nicht ganz problemlos. Seit fast 20 Jahren stellt das STL-Format eine de facto Standardschnittstelle zu RP-Anwendungen dar. Trotz offensichtlicher Unzulänglichkeiten dieses Schnittstellenformats und zahlreicher Neuentwicklungen zur fehlerfreien Übertragung der CAD-Geometrie an die RP-Anlagen erweist sich ein Formatwechsel derzeit als nahezu unmöglich. In diesem Vortrag werden die Ursachen dieser Schnittstellenproblematik und mögliche Alternativen einer verlustfreien Datenübertragung aufgezeigt. Ein Vergleich verschiedener Datenformate hinsichtlich der Erfüllung gesteigerter Anforderungen (z.B. Integration von Farb- und Texturinformationen, höhere Genauigkeiten, etc.) soll die Notwendigkeit einer verbesserten und vor allem standardisierten Schnittstelle verdeutlichen. Dabei wird insbesondere auf die Eignung derzeit verwendeter Schnittstellenformate für RP-Anwendungen eingegangen. Der Vortrag soll sowohl Konstrukteure als auch RP-Anwender für die Schnittstellenproblematik und die längst überfällige Definition eines einheitlichen Schnittstellenformats sensibilisieren. Ein möglicher Ansatz ist die Erweiterung des STEP-Formats um ein Anwendungsprotokoll für Anwendungen der RP-Technologie.

Keywords Rapid Prototyping, Schnittstellenformate, Schnittstellenproblematik

URN: urn:nbn:de:0009-2-15096

1. Einleitung

Die Globalisierung der Märkte und der damit einhergehende Wettbewerbsdruck auf die Unternehmen haben dazu geführt, dass die Umgehung des Wettbewerbs durch Konzentration auf Marktnischen nur noch selten zum Erfolg führt. Stattdessen gilt es, mit Hilfe geeigneter Simulations- und Verifizierungstechniken moderne Produktentwicklungsstrategien anzuwenden, die zu einer Technologieführerschaft beitragen können. Mit Hilfe des Rapid Prototyping (RP) hergestellte Bauteile weisen hierbei die größten Potenziale für die physische Produktverifikation auf, welche neben zur Herstellung von Design-, Geometrie- (Concept Modeling) und Funktionsprototypen (Functional Prototyping), zunehmend für die Fertigung von Vor- und Kleinserienwerkzeugen (Rapid Tooling) genutzt werden. Darüber hinausgehende Entwicklungen beschäftigen sich mit der Nutzung der RP-Technologie zur werkzeuglosen Fertigung mit seriennahen Eigenschaften (Direct Manufacturing), wobei der derzeitige Einsatz noch auf Nischenanwendungen in der Kleinserien-Fertigung beschränkt bleibt.

Gleichzeitig neben der rasant fortschreitenden Entwicklung der generativen Fertigungsverfahren bereitet die Anbindung dieser Technologie als integraler Bestandteil des Produktentwicklungsprozesses an die CAx-Landschaft nach wie vor Schwierigkeiten. Aus datenflusstechnischer Sicht bildet die Grundlage jeder RP-Anwendung eine geschlossene Form der Beschreibung der herzustellenden Geometrie, wobei die notwendigen Informationen den unterschiedlichsten Quellen entnommen werden können (vgl. Abb. 1).

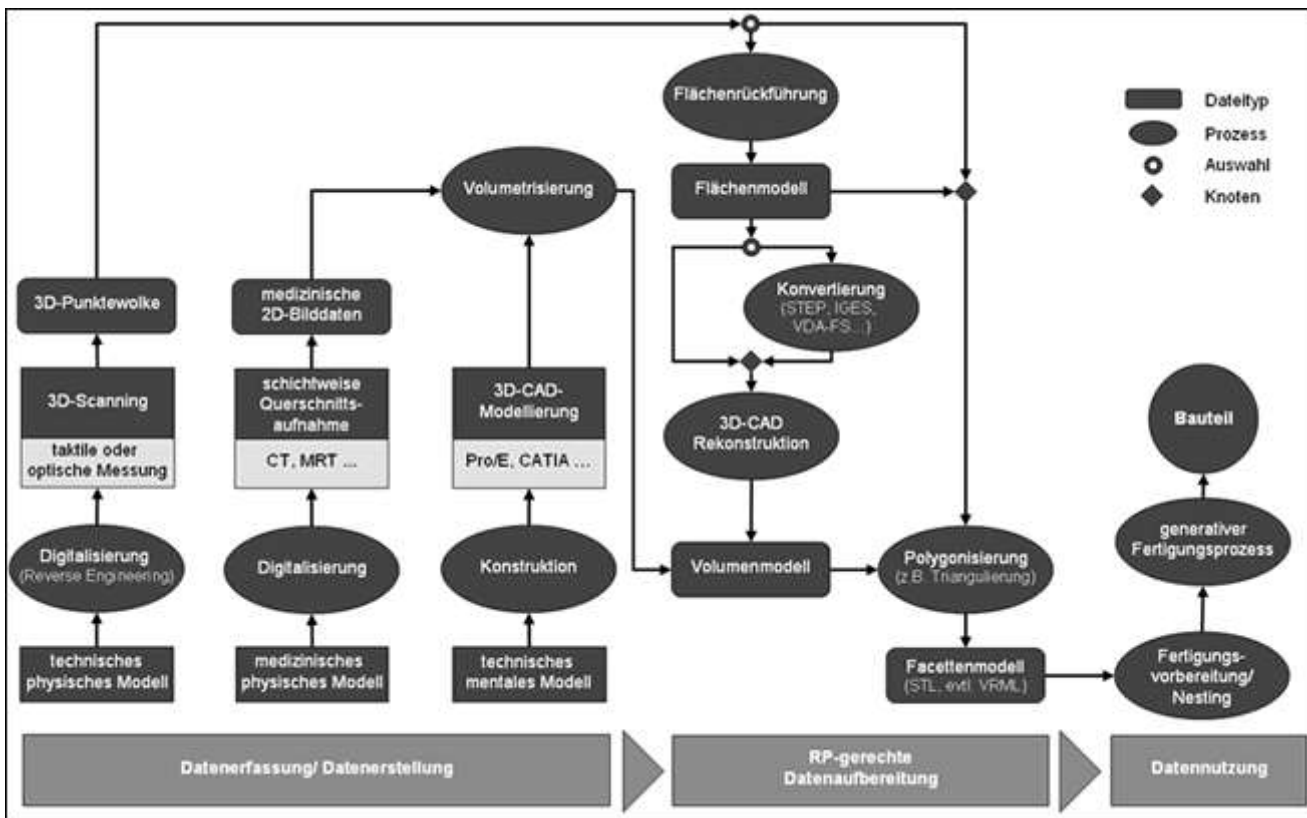


Abb. 1: Datentransfer der Konstruktionsdaten für RP-Anwendungen

So entsteht je nach Verfahren zur Datenerfassung bzw. Datenerstellung (3D-Scanning im Sinne des Reverse Engineering, schichtweise Querschnittsaufnahmen oder 3D-CAD-Modellierung) mit mehr oder weniger hohem Aufwand an Datenaufbereitung eine geeignete Repräsentationsform des Prototypen (Flächen- bzw. Volumenmodell). Diese gilt es verlustfrei an RP-Anwendungen zu übergeben.

2. RP-gerechte Datenaufbereitung

Der gesamte Prozess des Rapid Prototyping basiert auf der Erzeugung von Schichtdaten der Geometrie, dem so genannten Slicing der Geometrierepräsentation zur Gewinnung der notwendigen Schichteninformationen, der Ermittlung der Pfade für das Laser-Scanning bzw. für den Materialauftrag und das anschließende Post Processing (Abb. 2). Hierbei haben sich im Wesentlichen zwei unterschiedliche Vorgehensweisen etabliert, das Slicing eines RP- (STL, CFL, VRML, PLY, ...) bzw. neutralen Schnittstellenformats (STEP, IGES, SAT, ...) sowie das Direct Slicing, also die unmittelbare Schichtzerlegung des nativ vorliegenden CAD-Modells. Beide Herangehensweisen haben ihre Vor- sowie Nachteile und damit in jedem Fall ihre Daseinsberechtigung. Jedoch existiert derzeit keine zufrieden stellende Lösung auf dem Markt, die die Bedürfnisse der Anwender zur vollsten Zufriedenheit erfüllt.

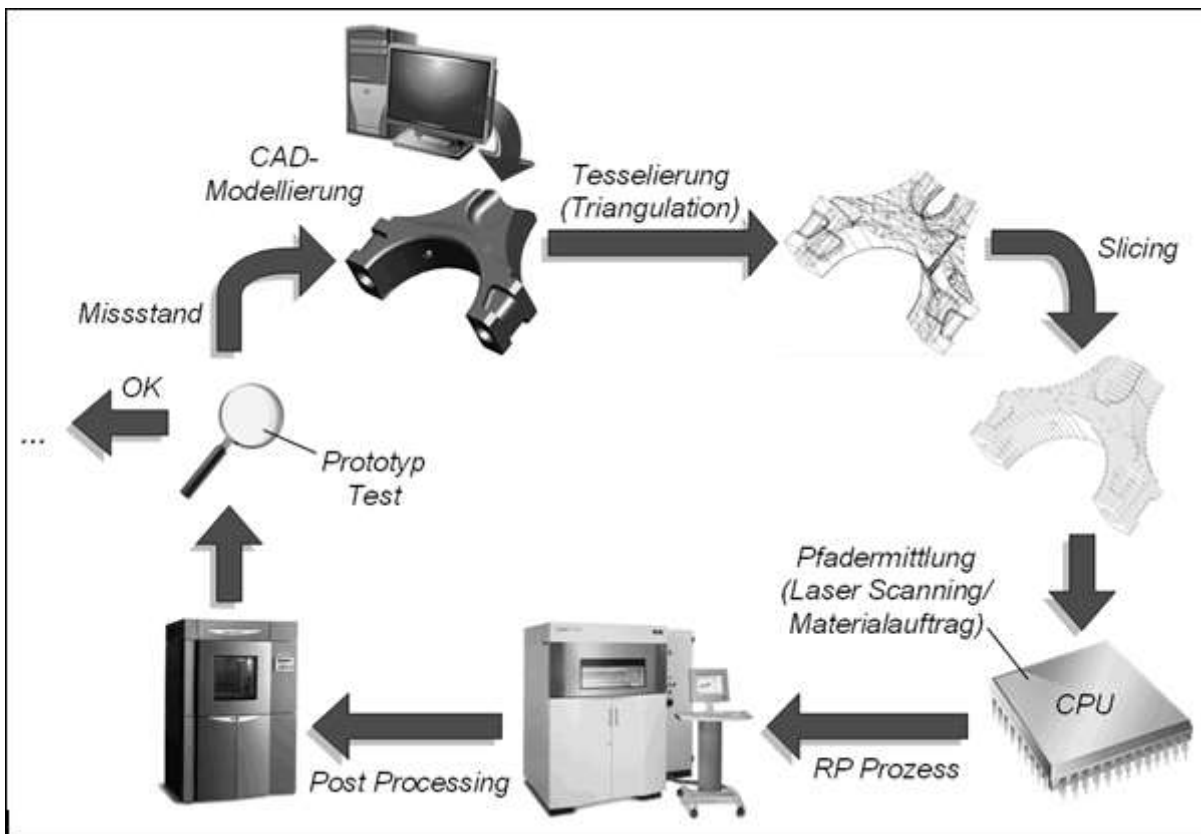


Abb. 2: Rapid Prototyping Prozesskette

2.1. STL als Standardschnittstelle

Die Ermittlung der notwendigen Schichteninformationen wird durch eine mathematische (virtuelle) Zerlegung der Geometrierepräsentation anhand eines Schnittstellenformats vorgenommen. Ein weit verbreitetes Verfahren ist die Nutzung des STL-Datenformats (STL – ursprünglich STereoLithography, heute Surface Tessellation Language, auch bekannt unter Standard Triangulation Language) als De-facto-Standard für Anwendungen der Rapid Technologie. Hierbei wird die Oberfläche eines 3D-Modells tesseliert, genauer gesagt trianguliert, also mit Dreiecken unterschiedlicher Größe überzogen. Einstellparameter (z.B. Sehnenhöhe) steuern die Größe der Dreiecksfacetten und somit die Genauigkeit der Approximation. Trotz bis dato fehlender Standardisierung dieser Schnittstelle hat sich das STL-Format mittlerweile in allen Bereichen der Rapid Technologie etabliert. Zum Einen durch die einfache mathematische Beschreibung der facettierten Oberfläche und zum Anderen durch die frühzeitige Veröffentlichung der Schnittstellenbeschreibung wurde die rasante Verbreitung dieses Dateiformats begünstigt, so dass nahezu jedes CAD-System und jede RP-Software die STL-Schnittstelle unterstützt. Doch gerade hier liegt ein wesentliches Problem, da die Anforderungen an Schnittstellenformate für RP-Zwecke gestiegen sind, während das STL-Format nicht oder nur kaum weiterentwickelt wurde [1]. Im Zuge der Verfahrensverbesserungen und Neuentwicklungen heutiger RP-Anwendungen haben sich folgende Nachteile des STL-Formats herauskristallisiert:

- STL stellt aufgrund der Polygondarstellung nur eine Approximation der Originalgeometrie dar. Die Möglichkeiten der Nachbesserungen am STL-Bauteil sind aufgrund der Abweichungen vom Originalmodell eingeschränkt.
- Die in CAD-Systemen umgesetzten Algorithmen zur Konvertierung in das STL-Format liefern keine konstant hohe Qualität. Hierbei kommt es häufig zu Umsetzungsfehlern und syntaktischen Darstellungsfehlern (Lücken, Überlappungen, Fehlorientierung der Normalenvektoren, etc.) [2]. Die hier zugrunde liegenden Facettierungsalgorithmen stammen noch aus der Zeit vor der generativen Fertigungstechnik und wurden hauptsächlich zur Visualisierung von Geometrien in der Computertechnik, insbesondere für die Spiele- und Grafikanwendungen entwickelt, bei denen keine Notwendigkeit von geschlossenen und richtig orientierten Oberflächen bestand. Dies macht häufig eine zum Teil sehr aufwändige Nachbearbeitung mit entsprechender RP-Software oder der Front-End-Software der RP-Anlage notwendig.

- STL-Dateien erreichen ein Vielfaches der ursprünglichen Dateigröße. Dieser Umstand ist auf die redundante Speicherung der Dreieckskoordinaten und des Normalenvektors zurückzuführen. Jede Dreiecksfacette wird durch die Speicherung ihrer drei Eckpunkte und des Normalenvektors definiert. Da immer mindestens drei, je nach Komplexität des Bauteils auch mehrere Dreiecke einen gemeinsamen Eckpunkt besitzen, wird jeder Punkt mindestens dreimal abgespeichert. Mit Hilfe der Rechte-Hand-Regel könnte schon zudem die Richtung des Normalenvektors ermittelt werden und macht eine explizite Angabe hinfällig. Die zunehmend erreichbare Genauigkeit (Auflösung) moderner RP-Anlagen wird die Dateigröße im STL-Format auch zukünftig weiter ansteigen lassen.
- Die für neuere RP-Verfahren geeigneten Informationen bezüglich Farbgebung, Texturen oder auch Materialien lassen sich nicht oder nur schlecht im STL-Format speichern und auch dann wäre diese Information mit dem gesamten Dreieck gekoppelt.
- Numerische Daten werden durch Datentypen mit unzureichender Genauigkeit dargestellt. Dies führt zu Rundungsfehlern bei Nachfolgeprozessen. Die Tatsache, dass Facettenkoordinaten immer positive Werte annehmen müssen (positiver Oktant), verstärkt diesen Effekt.
- Das anschließende Slicing des facettierten Körpers kann sich bei großen STL-Dateien aufgrund fehlender Topologie als sehr zeitaufwändig erweisen.

Diese Unzulänglichkeiten sowie die Tatsache, dass selbst bei expliziter Angabe der Einstellparameter für die Feinheit der Triangulierung und somit letztendlich für die Genauigkeit der Approximation größere Abweichungen nach dem Slicing-Prozess auftreten können (Abb. 3), genügt aus heutiger Sicht nicht mehr den gestiegenen Anforderungen der RP-Technologie.

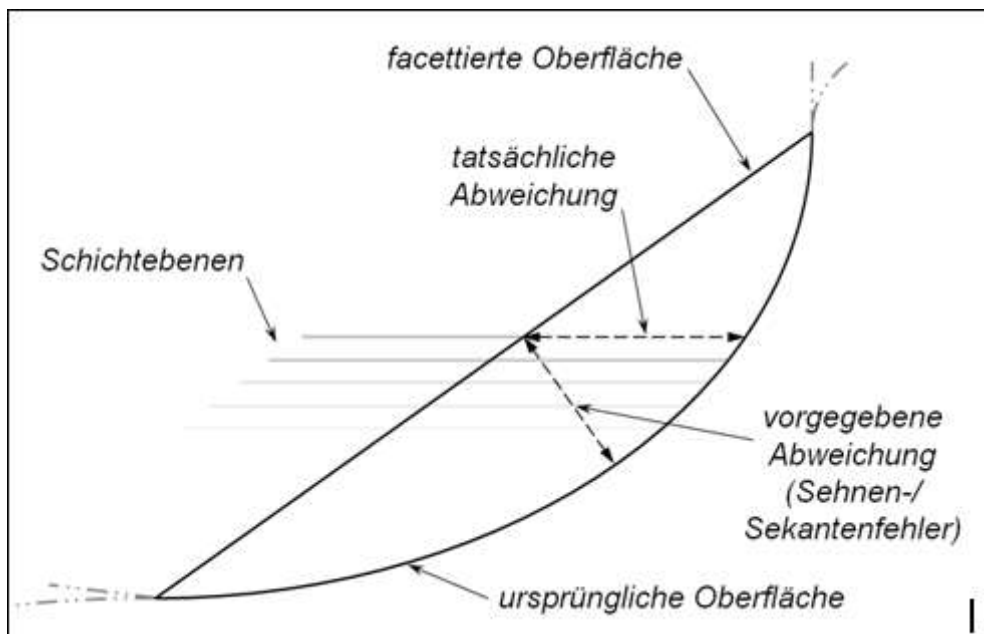


Abb. 3: Tesselierungs- und Schichtzerlegungsfehler

2.2. Weiterentwicklungen des STL-Formats

Die offensichtlichen Schwächen des STL-Formats wurden in den letzten Jahren erkannt, so dass einige Weiterentwicklungen vorangetrieben wurden [3]. So präsentieren WU und CHEUNG ein erweitertes STL-Format (enhanced STL), bei dem Dreiecksflächen durch Tetraeder zur Erhöhung der Approximationsgenauigkeit ersetzt werden (siehe Abb. 4) [4].

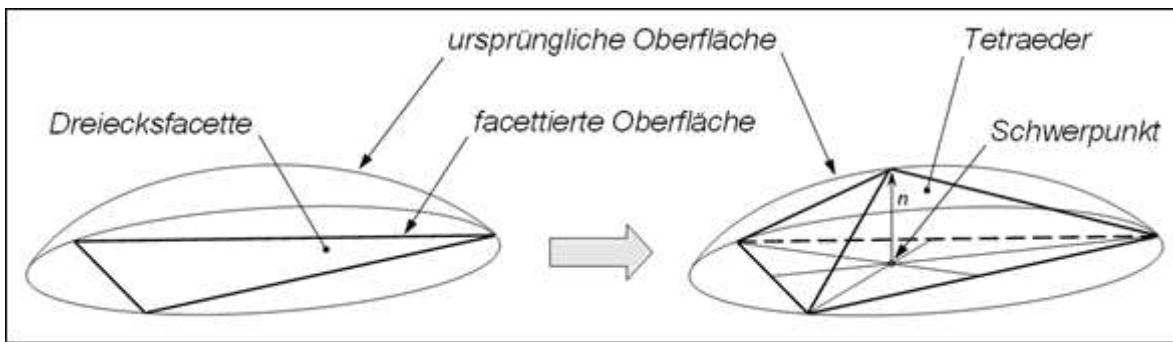


Abb. 4: Erhöhung der Approximationsgenauigkeit durch Tetraeder-Bildung

Hierzu werden zwei bis dato nicht verwendete Bytes im STL-Format mit zusätzlichen Informationen belegt, deren Auswertung für die Erzeugung der Tetraeder notwendig ist. Dies sind zum Einen die Höhe des Tetraeders und zum Anderen eine Information z.B. bezüglich Farbgebung oder Markierung. Mit diesem Verfahren wird im Ansatz den gestiegenen Anforderungen genüge getan, jedoch bleibt zum Beispiel das Problem der redundanten Speicherung und somit des unnötig hohen Datenvolumens bestehen. Eine weitere Problematik ergibt sich beim Slicen des Modells. Im Gegensatz zum STL-Format werden hier Tetraeder geschnitten, so dass sich ein genauere Polygonzug ergibt, jedoch mehr Rechenaufwand betrieben werden muss. Des Weiteren können Zickzack-Verläufe auftreten (vgl. Abb. 5), die es in einem zusätzlichen Optimierungsschritt zu eliminieren gilt um den Konturzug zu glätten. WU und CHEUNG beziffern den Annäherungsfehler beim Slicen von Tetraedern im Vergleich zum herkömmlichen STL auf etwa ein Viertel.

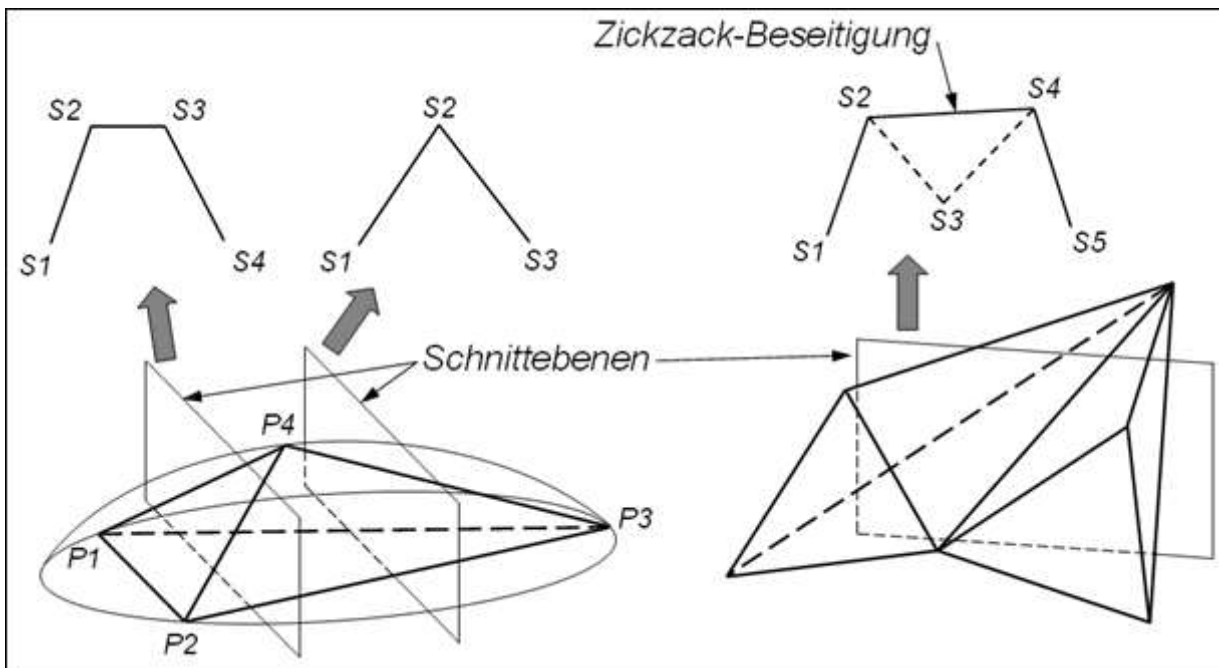


Abb. 5. Verschiedene Slicing-Situationen an Tetraedern

Ein bereits Anfang der 90er Jahre entwickeltes Format ist das am Rensselaer Polytechnic Institute ins Leben gerufene RPI-Format (Rapid Prototyping Interface) [5]. Aufgrund einer bestehenden Aufwärtskompatibilität kann das STL-Format in die RPI-Struktur überführt werden, was die Durchsetzbarkeit einer flächendeckenden Verbreitung begünstigt. Die Stärken des RPI-Formats liegen in der Flexibilität durch die Möglichkeit topologische Informationen an einzelne Dreiecksfacetten anzuhängen sowie in der Erweiterbarkeit des Formats. Der Vorteil der Flexibilität ist gleichzeitig auch ein entscheidender Nachteil. Dies kann nachfolgende Prozesse wie das Slicen zu einem aufwändigen und komplexen Vorgang machen. Ebenso müssen die topologischen Informationen mit einem erhöhten Rechneraufwand aus den unsortierten Facettendaten des STL-Formats abgeleitet werden. Die Speicherung der Dreiecksdaten an sich erfolgt jedoch redundanzfrei. Des Weiteren ermöglicht das RPI-Format die Darstellung sowohl facettierter Körper als auch reiner CSG-Solids (Constructive Solid Geometry).

CHUA et al entwickelten das LMI-Format (Layer Manufacturing Interface), welches ebenfalls als Weiterentwicklung des STL-Formats zu verstehen ist [6]. Die Grundlage bildet das STL-Format, wobei durch Umstrukturierung die Redundanz vermieden wird und durch das Hinzufügen topologischer Informationen eine B-Rep-Darstellung (B-Rep – Boundary Representation) sowohl facettierter als auch präziser Modelle möglich ist. Die Anreicherung mit topologischen Informationen vereinfacht zudem die

Überprüfung der Modellkonsistenz aber auch nachfolgende Prozesse, wie z.B. die Schichtzerlegung. Beim Slicing muss für jede einzelne Schicht ermittelt werden, welche Facetten geschnitten werden. Ohne Topologieinformationen muss für jede Schicht jedes einzelne Dreieck überprüft werden. Dies führt zu einer zeitintensiven Rechnung.

Verbesserungen am STL-Format, wie die Erhöhung der Darstellungsgenauigkeit oder die Speicherung von Farbinformationen für farbige Prototypen, werden in aller Regel zu einer Verfeinerung der Vernetzung und somit zum Anstieg der Dateigröße führen. Um dieser Problematik gerecht zu werden, entwickelten ZHANG et al das CS-Format (Compressed STL). Die üblichen 32-Bit-Integer (ermöglicht die Speicherung von 4 Milliarden Eckpunkten) werden bei diesem Ansatz durch 14-Bit- oder 20-Bit-Integer ersetzt, so dass die Speicherung überflüssiger Leerstellen entfällt. Dies führt zu erheblich geringeren Dateigrößen. Da aber im Grunde genommen auch hier ein STL-Format vorliegt, treten auch hier die zuvor genannten Unzulänglichkeiten auf, so dass sich neben der Ersparnis an Speicherbedarf keine weiteren Vorteile ergeben.

2.3. Weitere 3D-Schnittstellenformate

Aufgrund der zuvor genannten Unzulänglichkeiten des STL-Formats haben sich in den letzten Jahren zahlreiche Schnittstellenformate weiterentwickelt, die je nach Anwendungsfall eine bessere Alternative darstellen können. Doch auch hier konnte sich bisher kein Format als universelles Schnittstellenformat für RP-Anwendungen durchsetzen. Eine bereits früh erkannte Alternative zum STL ist das VRML-Format (Virtual Reality Modeling Language). Dieses Format speichert Farb- und Texturinformationen, kann komplexe Geometrien darstellen und weist keine redundante Speicherung der Daten auf. Doch auch dieses Format wurde ursprünglich für weborientierte virtuelle 3D-Welten entwickelt, so dass es aufgrund der Komplexität ihrer Struktur für RP-Applikationen, insbesondere für die Schichtzerlegung, nicht optimiert ist und lediglich Flächen, jedoch nicht Körper (Solids) darstellen kann. Dies führt dazu, dass das VRML-Format sehr gut für die Erzeugung farbiger Prototypen geeignet ist^[7], jedoch für zukünftige Anforderungen hinsichtlich der Solid-Repräsentation nicht ausreichend ist. Dennoch weist dieses Format gute Eigenschaften auf, so dass es seit langem als mögliche Standard-Schnittstelle für RP-Anwendungen zur Diskussion steht. Da aber VRML-Dateien aufgrund ungeeigneter Konverter in CAD-Systemen oft fehlerbehaftet sind, ist derzeit eine Substitution des STL-Formats durch VRML unwahrscheinlich.

Ein an der Stanford Universität entwickeltes Austauschformat ist das PLY-Format (Polygon File Format), auch bekannt als Stanford Triangle Format. Dieses Format, hauptsächlich für den Austausch graphischer 3D-Objekte aus Scandaten entwickelt, beschreibt ein Objekt als Sammlung von Punkten, Flächen sowie weiterer Elemente mit gespeicherten Eigenschaften wie Farbinformationen, Flächennormale, Transparenz, Texturkoordinaten, etc. Vom Aufbau her gleicht die Struktur der einer VRML-Datei. Eine Liste (vertex list) beinhaltet Koordinaten-Tripel der Dreieckspunkte (vertices), eine zweite Liste (facet list) beschreibt die Facetten durch Verweise auf die Vertex-Indizes der Vertex-Liste. Somit werden wie bei VRML die Daten redundanzfrei abgespeichert und halten die Dateigröße niedrig. Das PLY-Format gestattet des Weiteren die Definition weiterer Elemente (z.B. Kanten, Zellen, Materialien). Aufgrund der einfachen Dateistruktur und der Möglichkeit der Speicherung verschiedener Eigenschaften eines Polygons (hier: Dreieck) ist das PLY-Format eine elegante Alternative zum STL- und VRML-Format. Während dieses Format bereits beim 3D-Printing Verwendung findet, konnte es sich aufgrund komplexen Post-Processings dennoch nicht flächendeckend durchsetzen.

Das von Cubital Ltd. entwickelte CFL-Format (Cubital Facet List) stellt eine polygon-basierte Repräsentation eines Objekts dar. Die einzelnen Facetten bestehen aus n-seitigen Polygonen und nicht zwangsläufig nur aus Dreiecken. Die Speicherung der Facettendaten erfolgt redundanzfrei, da, ähnlich wie bei VRML und PLY, Facetten und Vertices getrennt gelistet werden. Die Beschreibung der Polygoneckpunkte erfolgt auch hier über Verweise auf indizierte Elemente der Vertex-Liste. Dies führt zu deutlich kleineren Dateigrößen als. Die Dateistruktur gestattet die Speicherung zusätzlicher benutzerdefinierter Informationen. Das Format ist simpel und erfüllt weitestgehend die gleichen Anforderungen wie STL, konnte sich aber auch bisher am Markt nicht durchsetzen.

Eine weitere aktuelle Entwicklung ist das von WANG et al vorgestellte ETL-Format^[8]. Dabei liegt der Fokus bei diesem Format auf der Darstellungsgenauigkeit, einer einfachen Schichtzerlegung sowie der Möglichkeit Farb- und Texturinformationen zu speichern. Das ETL-Format besteht aus zwei Teilen: eine Datei zur Beschreibung der Modellgeometrie (GDF – Geometry Description File) sowie eine Datei zur Beschreibung von Oberflächendetails wie Farbe und Textur (TMDF – Texture Mapping Description File). Dies führt jedoch zu einem ungeeigneten Handling beim Datentransfer. Des Weiteren konnte auch hier die

Dateigröße nicht signifikant verringert werden. Das Durchsetzungsvermögen dieses Dateiformats als zukünftige Standard-Schnittstelle für RP-Anwendungen bleibt fraglich.

Ähnliche Überlegungen zur Reduzierung des Speicherbedarfs führten im Rahmen der OpenRP-Initiative zum RP-Format (openRP File Format). Neben der enormen Komprimierung von bis zu 95% und einer Möglichkeit zur Verschlüsselung bietet aber auch dieses Format keine nennenswerten Vorteile gegenüber dem STL-Format. Weitere Formate, die zwar zu einer deutlichen Ersparnis des Speicherplatzbedarfs führen aber die gleichen Probleme wie das STL-Format aufweisen, sind z.B. das von Brock Rooney & Associates entwickelte STH-Format (Surface Triangles Hinted Format) oder auch das von MCMAINS vorgestellte SIF-Format (Solid Interchange Format).

2.4. Neutrale Schnittstellenformate

IGES (Initial Graphics Exchange Specification) stellt seit 1981 einen internationalen Standard für den Austausch von Graphikinformationen zwischen CAD-Systemen dar. IGES deckt sowohl CSG- als auch B-Rep-Repräsentationen ab und wird von zahlreichen RP-Systemen akzeptiert. Dieses Format erfreut sich eines hohen Bekanntheitsgrads und enthält alle Elemente für eine exakte Repräsentation des CAD-Modells. Ein gravierender Nachteil ist jedoch eine redundante Informationsspeicherung und eine fehlende Unterstützung facettierter Darstellungen. Zudem ist IGES durch die sequentielle Verarbeitung bei großen Dateien sehr langsam.

STEP (Standard for the Exchange of Product Data) ist ein internationaler Standard (ISO 10303) für den Austausch von Produktlebenszyklusdaten, d.h. neben der Beschreibung physikalischer Daten berücksichtigt dieses Format funktionale Aspekte aus allen Bereichen der Produktentwicklung [9]. STEP ist ein weltweit etabliertes und bewährtes neutrales Datenaustauschformat, welches einer stetigen Weiterentwicklung unterliegt. So werden für die verschiedensten Anwendungsbereiche Applikationsprotokolle bereitgestellt. Derzeit existiert kein Anwendungsprotokoll für die schichtweise Fertigung, wobei jedoch diese Bestrebungen vor einigen Jahren initiiert wurden und noch bis heute andauern. Bereits jetzt erfüllt STEP eine Vielzahl an Anforderungen eines RP-gerechten Datenaustauschs durch die Bereitstellung entsprechender Bausteine (IR – Integrated Resources, AIC – Application Interpreted Constructs) [9 , 10 , 11 , 12]. Tabelle 1 zeigt bereits in STEP enthaltene Ressourcen, die die wesentlichen Anforderungen der RP-Technologie erfüllen.

Anforderungen	Ressourcen in ISO 10303
<i>Geometrie & Topologie</i>	<i>Part 42 (Geometric and topological representation)</i> <i>AIC 511 (Topologically bounded surface)</i> <i>AIC 514 (Advanced boundary representation)</i>
<i>Materialinformationen</i>	<i>Part 45 (Materials)</i>
<i>Toleranzen</i>	<i>Part 47 (Shape variation tolerances)</i> <i>AIC 519 (Geometric tolerances)</i>
<i>Prozessinformationen</i>	<i>Part 49 (Process structure and properties)</i>
<i>mathematische Konstrukte</i>	<i>Part 50 (Mathematical constructs)</i>

Tab 1: RP-gerechte Ressourcen in STEP

STEP bietet prinzipiell die Möglichkeit ebenfalls Schichtdaten speichern zu können. So können bereits bewährte Teile der STEP-Ressourcen, wie z.B. IR42 für die Definition von Konturen und Füllmuster, IR45 für Materialinformationen oder auch IR49 für die Übertragung von Prozessinformationen, für einen RP-gerechten Datenaustausch verwendet werden. Einziger Nachteil des STEP-Formats ist, dass Informationen zum Teil redundant abgespeichert werden und STEP unzählige Informationen beherbergt, die für den RP-Prozess uninteressant sind. Dies erfordert spezielle Interpreter für den Austausch von Daten zwischen CAD- und RP-Systemen.

3. Schlussfolgerungen

In den letzten 15 Jahren wurden zahlreiche Weiterentwicklungen am STL-Format durchgeführt sowie Konzepte eines alternativen Schnittstellenformats entwickelt. Diese führten zu mehr oder weniger beträchtlichen Verbesserungen gegenüber dem STL-Format, konnten sich aber bis dato nicht etablieren. Des Weiteren haben die rasanten Entwicklungen der Rapid Technologie dazu geführt, dass mit zusätzlichen Modellinformationen eine Optimierung der einzelnen Prozesse möglich wäre. Gerade eine Automatisierung der Prozessplanung benötigt jedoch mehr geometrische und topologische Informationen [1 , 13]. Das rein geometrische STL-Format kann das nicht mehr bieten, wohingegen einige neue Ansätze die Übertragung topologischer Informationen, Prozess-, Farb- oder Materialinformationen berücksichtigen (vgl. Tabelle 2).

		Schnittstellenformate												
		CFL	CS	enh.STL	ETL	LMI	PLY	RP	RPI	SIF	STEP	STH	STL	VRML
Anforderungen	Darst. allg. Flächen					■					■			
	allg. Facettendarst.	■									■			
	triangulierte Darst.	■	■	■	■	■	■	■	■	■	■	■	■	■
	B-Rep					■			■		■			
	CSG								■		■			
	Farbinformationen	■		■	■		■				■			■
	Materialinformationen	■					■				■			■
	Prozessinformationen	■					■				■			■
	redundanzfrei	■				■	■	■	■	■		■		■
	erweiterbar	■					■		■		■			
	geringer Speicherbedarf	■	■			■	■	■	■	■		■		■
	hoher Verbreitungsgrad			■			■				■		■	■
	verbessertes Slicing		■		■	■		■		■	■	■	■	
	Toleranzen										■			

Tab. 2: Erfüllung der Anforderungen verschiedener Formate

Die unüberschaubare Zahl an Entwicklungen einer vermeintlich RP-gerechten Datenschnittstelle trägt zur Verunsicherung der Anwender bei und verhindert den Wunsch nach Veränderungen. Trotz allem scheint bei näherer Betrachtung das LMI-Format eine verheißungsvolle Lösung darzustellen. Gerade durch die aufgrund der Topologie ermöglichte Darstellung sowohl facettierter als auch präziser Modelle sowie die Einsparung von Speicherplatz, erscheint zunächst dieses Format prädestiniert für eine Substitution des STL-Formats. Der vielleicht vielversprechendste Ansatz ist jedoch wohl die Entwicklung eines STEP-Anwendungsprotokolls für Anwendungen des Rapid Prototyping. STEP bietet die günstigsten Voraussetzungen für einen verlustfreien Datenaustausch mit umfangreichen Produkt- und Prozessinformationen sowie der Möglichkeit einer Beschreibung sowohl facettierter Darstellungen als auch einer exakten Geometriebeschreibung [10 , 11 , 12]. Dies führt dazu, dass ein als STEP gespeichertes Modell bereits im CAD-System optimal orientiert und in Schichten zerlegt werden kann [14] . Sofern dies nicht erwünscht ist, kann das Modell in einem neutralen Format zu RP-Systemen übertragen werden. Des Weiteren könnte STEP Schichtdaten speichern, die zwischen CAD- und RP-System oder von RP- zu RP-System transferiert werden können. Die weite Verbreitung und die Tatsache, dass STEP in vielen Bereichen seit Jahren intensiv und erfolgreich zum Einsatz gekommen ist, begünstigt den Vorstoß, STEP um Aspekte des Rapid Prototyping zu erweitern und somit als anerkannter Standard das bislang favorisierte STL-Format abzulösen.

4. Ausblick

Der wichtigste Schritt ist wohl zunächst die Erstellung RP-relevanter Empfehlungen und Richtlinien, die eine Grundlage für die Bildung einer eigenen nationalen oder sogar internationalen Normenreihe darstellen können. Für eine erfolgreiche Substitution des bislang verwendeten STL-Formats durch ein erweitertes STEP-Format müssen sowohl seitens der RP-Anwender als auch der CAD- und RP-Systemhersteller entsprechende Vorkehrungen bzw. Voraussetzungen geschaffen werden. Neben einer breiten Anwenderakzeptanz trägt auch die Integration notwendiger Translators in der CAD- und RP-Software zur Unterstützung des verbesserten STEP-Formats zu einer flächendeckenden Verwendung dieser RP-gerechten Schnittstelle bei. Exemplarische Umsetzungen werden des Weiteren notwendig sein um Anwender von der Notwendigkeit eines Formatwechsels zu überzeugen. Denn auch hier gilt unter den RP-Anwendern nach wie vor: If it's not broken, don't fix it.

5. Literatur

- [1] Bohn, J. H.: File Format Requirements for the Rapid Prototyping Technologies of Tomorrow, International Conference on Manufacturing Automation, University of Hong Kong, Hong Kong, 1997, S. 878-883.
- [2] Nagy, M. S.; Matyasi, G. Y.: Analysis of STL files, Mathematical and Computer Modelling, 38, 2003, S. 945-960.
- [3] Stroud, I.; Xirouchakis, P. C.: STL and Extensions, Advances in Engineering Software, 31 (2), 2000, S. 83-95.
- [4] Wu, T.; Cheung, E. H. M.: Enhanced STL, International Journal of Advanced Manufacturing Technology, 29 (11-12), 2006, S. 1143-1150.
- [5] Rock, S. J.; Wozny, M. J.: A Flexible File Format for Solid Freeform Fabrication, Solid Freeform Fabrication Symposium Proceedings, University of Texas, Austin, TX, 1991, S. 1-12.
- [6] Chua, C. K.; Gan, G. K. J.; Tong, M.: Interface between CAD and Rapid Prototyping Systems - Part 2: LMI - an Improved Interface, International Journal of Advanced Manufacturing Technology, 13, 1997, S. 571-576.
- [7] Ming, L. W.; Gibson, I.: Specification of VRML in Color Rapid Prototyping, International Journal of CAD/CAM, 1 (1), 2002, S. 1-9.
- [8] Wang, D.-X.; Leng, H.-W.: Slicing of CAD Models in ETL Format, Rapid Prototyping Journal, 13 (1), 2007, S. 53-57.
- [9] International Organization for Standardization: Industrial Automation Systems and Integration – Product Data Representation and Exchange – Part 1: Overview and fundamental principles, ISO 10303-1, 1994.
- [10] Kumar, C. et al.: Towards STEP-Based Data Transfer in Layered Manufacturing, International IFIP, The Globalization of Manufacturing in the Digital Communications Era of the Tenth proceeding, 1998.
- [11] Kobayashi, K. G.; Fujii, M.; Prinz, F. B.: Conceptual Data Model for Advanced Rapid Prototyping, Initiatives of Precision Engineering at the Beginning of a Millennium, Springer US, 2002, S. 122-126.
- [12] Patil, L. et al.: A Proposed ISO 10303 (STEP)-based Approach for Representing Heterogeneous Objects for Layered Manufacturing, Rapid Prototyping Journal, 8 (3), 2002, S. 134-146.
- [13] Pratt, M.; Marsan, A. L.; Kumar, V.; Dutta, D.: An Assessment of Data Requirements and Data Transfer Formats for Layered Manufacturing, NISTIR 6216, National Institute of Standards and Technology, Gaithersburg, MD, 1998.
- [14] Zhou, M. Y.; Xi, J. T.; Yan, J. Q.: STEP-Based Direct Slicing Algorithm for Rapid Prototyping, Shanghai Jiaotong Daxue Xuebao, 37 (5), 2003, S. 737-740.

6. Kontaktangaben

Dipl.-Ing. Stéphane Danjou
Universität Duisburg-Essen Institut für Produkt Engineering / IPE Produktionstechnologie und
Produktentwicklung
Lotharstraße 1
47057 Duisburg / Germany
Tel.: +49 203/379-3935
Fax: +49 203/379-4379
Email: stephane.danjou@uni-due.de
WEB : www.uni-due.de

Prof. Dr.-Ing. Dipl.-Math. Peter Köhler
Universität Duisburg-Essen Institut für Produkt Engineering / IPE Produktionstechnologie und
Produktentwicklung
Lotharstraße 1
47057 Duisburg / Germany
Tel.: +49 203/379-2790
Fax.: +49 203/379-4379
Email: peter.koehler@uni-due.de
WEB: www.uni-due.de

Volltext

Lizenz

Jedermann darf dieses Werk unter den Bedingungen der Digital Peer Publishing Lizenz elektronisch übermitteln und zum Download bereitstellen. Der Lizenztext ist im Internet unter der Adresse http://www.dipp.nrw.de/lizenzen/dppl/dppl/DPPL_v2_de_06-2004.html abrufbar.

Empfohlene Zitierweise

Danjou S, Koehler H (2008). Vorbereitung von CAD-Konstruktionsdaten für den RP-Einsatz – eine Schnittstellenproblematik. RTejournal - Forum für Rapid Technologie, Vol. 5(2008). (urn:nbn:de:0009-2-15096)

Bitte geben Sie beim Zitieren dieses Artikels die exakte URL und das Datum Ihres letzten Besuchs bei dieser Online-Adresse an.