

# **Einführung von offenen Standards in der Ablaufsimulation am Beispiel der Schnittstelle Layout-Fördertechnikplanung**

## ***Introducing Open Standards to Material Flow Simulation***

Gottfried Mayer, Marielouise Mieschner, BMW AG, München (Germany),  
gottfried.mayer@bmw.de, marielouise.mieschner@bmw.de

**Abstract:** The quality of material flow simulation results depends on the quality of the used input data. Therefore it is important to import complex data via an automated interface to avoid mistakes. AutomationML is an open data exchange format that can be used to hand over data from different fields of the digital factory. This article describes the data exchange from CAD conveyor planning to the material flow simulation by using AutomationML.

## **1 Motivation**

Daten sind der Grundstein jeder Ablaufsimulation. Um homogen auf Planungsdaten zugreifen zu können, sind in der Vergangenheit diverse Schnittstellen zwischen den Datenbanken und den Simulationssystemen entstanden. Meist wurden hier proprietäre Datenformate zum Austausch gewählt, die exakt die zu übergebenden Daten abbilden. Das hat den Vorteil, dass die Dateien klein sind. Es hat aber den Nachteil, dass die Schnittstelle ausschließlich für den gewählten Anwendungsfall funktioniert. Muss das Datenmodell geändert oder erweitert werden, ist dies meist schwierig, da sowohl das abgebende als auch das aufnehmende System geändert werden müssen.

Diese Lücke versucht das AutomationML-Konsortium zu schließen (vgl. AutomationML 2017). Der Ansatz, verschiedene standardisierte Datenformate zu einem Austauschformat für die Daten der Automatisierungstechnik zusammenzufassen, scheint auch für die digitale Fabrik hilfreich zu sein (vgl. Drath 2010). Um dies für die Automobilindustrie zu überprüfen und zu strukturieren, wurde innerhalb des VDA-Arbeitskreises *Digitale Fabrik* die Projektgruppe *AutomationML* gegründet (VDA: Verband der Automobilindustrie). Ziel dieser Gruppe ist es, eine VDA-Empfehlung zu erarbeiten, die die Rahmenbedingungen absteckt, sodass AutomationML (AML) projektübergreifend eingesetzt werden kann. Um die erarbeiteten Theorien zu überprüfen, wurde beschlossen, die Schnittstelle GSL-FT (Generische Simulationslösung – Fördertechnik) (vgl. Jörg und Wacker 2008)

zwischen der Layout-Fördertechnikplanung und der Ablaufsimulation in AML zu migrieren.



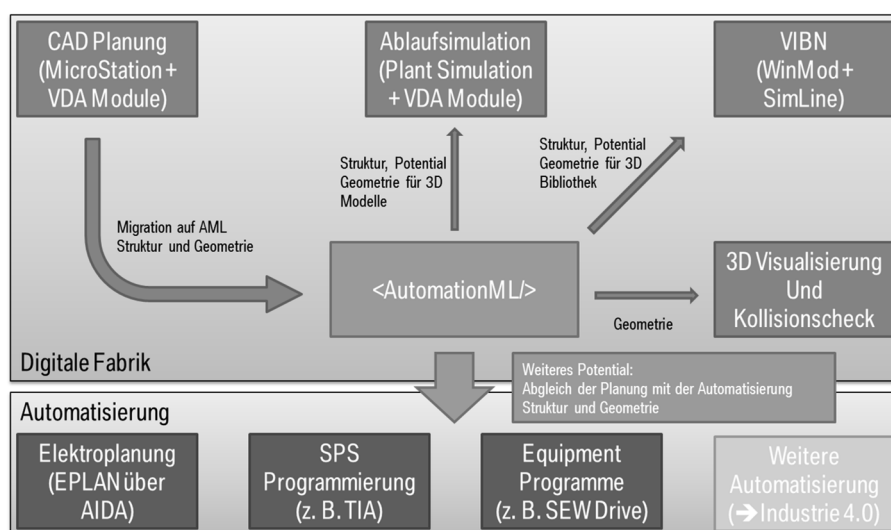
**Abbildung 1:** *Proprietäre Schnittstelle GSL-FT*

Diese GSL-FT-Schnittstelle (Abb. 1) wurde geschaffen, um den Simulationsexperten das Nachbauen und Parametrisieren des Fördertechniklayouts im Simulationstool zu ersparen und dies zu automatisieren. Hierzu muss im Quellsystem, in diesem Fall MicroStation mit den TriCAD Fördertechnikmodulen, das vorhandene Fördertechniklayout um die Flussrichtung erweitert werden. Ist diese vollständig, wird eine XML-Datei (XML: extensible Markup Language) erzeugt, die die Art der Fördertechnik, deren Koordinaten, Geschwindigkeiten, Vorgänger und Nachfolger sowie Verfügbarkeitsdefinitionen der Komponenten beinhaltet.

Diese Daten können nun direkt in ein Modell überführt oder über eine Zwischenoberfläche (GSL-FT-Modul im Simulationsassistenzsystem SimAssist) nochmals bearbeitet bzw. verifiziert und validiert werden (zur Verifikation und Validierung vgl. Rabe 2008) werden. Der Zwischenschritt dient dazu, um bekannte Daten, die im Quellsystem nicht vorhanden sind, wie z. B. spezielle Bahngeschwindigkeiten, noch vor dem Erzeugen des Modells hinzufügen zu können.

Diese Schnittstelle wurde im Jahr 2009 von den VDA-Arbeitsgruppen *Ablaufsimulation* und *Digitale Fabrikplanung* definiert, eingeführt und ist seitdem benutzbar. Da in den Daten die Struktur der Fördertechnik auslesbar ist, kam in den letzten Jahren immer mehr Interesse an diesen auf. Als Beispiel sei hier die virtuelle Inbetriebnahme (VIBN) zu nennen. In der VIBN werden die Programme für die Steuerung der Anlagen vorab getestet. Dafür muss die komplette Anlage inklusive deren Sensorik abgebildet werden. Es wurden Versuche gemacht, hier auch die GSL-FT-Schnittstelle als Basis zu verwenden. Die ersten Ergebnisse waren vielversprechend; es gibt aber weiteren Bedarf, das Datenformat um Informationen zu erweitern, wie z. B. die geometrische Darstellung der Objekte.

Nach reichlicher Diskussion und Überlegung wurde beschlossen, das vorhandene GSL-FT-Format nicht zu erweitern, sondern die Schnittstelle in AML zu überführen. Neben der einfacheren Erweiterbarkeit, die durch das Format selbst gegeben (siehe Abschnitt 2) ist, war auch die weitere Verwendbarkeit (Abb. 2) eine ausschlaggebende Begründung für diese Entscheidung.



**Abbildung 2:** Verwendbarkeit der neuen Schnittstelle

Durch die Erweiterung des Datenaustauschformates um die 3D-Geometrie entsteht für die Ablaufsimulation die Möglichkeit, direkt aus der Schnittstelle heraus dreidimensionale Modelle zu erstellen. Genau dieselbe Datenbasis wird dann für die virtuelle Inbetriebnahme verwendet, um dort die Sensorik-Modelle zu erstellen. Ein weiterer Anwendungspunkt aus der digitalen Fabrik ist dann noch die 3D-Visualisierung inklusive Kollisionscheck, für die diese Daten direkt weiterverwendet werden können.

Ein zusätzliches Potenzialfeld für die Fördertechnikdaten im AML-Format sind die Themen der Automatisierung. Um z. B. ein SPS-Programm (SPS: speicherprogrammierbare Steuerung) für die Anlage erstellen zu können, benötigt man die Konfiguration der Hardware, die beschreibt, welches Equipment vorhanden ist. Dies kann aus der AML-Datei ausgelesen und weiterverwendet werden. Das gleiche gilt für die Planung der Elektroinstallation oder die Programmierung des Equipments selbst.

## 2 Aufbau AML

AML ist keine komplett neue Erfindung eines Datenformates. Vielmehr wurde hier auf vorhandene Standardformate zurückgegriffen, die auf XML basieren. Kernstück bildet das Dateiformat CAEX (Computer Aided Engineering Exchange), ein Format zur Speicherung und Darstellung von hierarchischen Strukturen.

In AML werden vier Strukturen unterschieden:

Die *Rollenklassenbibliothek* dient dazu, eine Sammlung an Objekten definieren zu können und diese auch voneinander abzuleiten. Der AML-Verein stellt eine Bibliothekssammlung (vgl. AutomationML 2017) zur Verfügung, die beliebig erweitert werden kann. Wichtig ist dabei, dass Bezüge zu den Basisklassen hergestellt werden.

Die *Schnittstellenklassenbibliothek* definiert Schnittstellen innerhalb der AML-Datei sowie externe Verweise.

Die *Systemklassenbibliothek* dient dazu, spezielle Ableitungen aus den Rollenklassen zu definieren und für die eigene Verwendung anzupassen. Hier können zusätzliche hersteller- oder anwenderspezifische Daten ergänzt werden.

In der *Instanzenhierarchie* werden final die einzelnen Objekte beschrieben. Diese müssen von den Klassenbibliotheken abgeleitet sein und repräsentieren die realen Objekte mit deren Eigenschaften.

Diese Struktur findet sich in allen AML-Dateien wieder und kann durch externe Definitionen erweitert werden. Diese externen Anhänge werden über definierte Schnittstellen angebunden (Schnittstellenklassenbibliothek). Aktuell werden hier Collada (vgl. Collada 2017) zur Beschreibung von Geometrie und Kinematik sowie PLCopen XML (vgl. PLCopen 2017) zur Verhaltensbeschreibung, genutzt.

### 3 VDA-Empfehlung

Da in offenen Formaten die Strukturen nicht fest definiert sein können, trat die Frage auf, wer definiert für die einzelnen Anwendungen die Struktur innerhalb von AML. Da eine solche Definition jegliche Software betrifft, die das Format später lesen oder schreiben können soll, wurde innerhalb des VDA-Arbeitskreises *Digitale Fabrik* eine Projektgruppe gegründet, die dieser Problematik nachgeht.

Bei der Betrachtung, welche Inhalte firmenübergreifend definiert werden sollen, wurde schnell klar, dass für diese Umfänge eine VDA-Empfehlung nötig sein wird. Diese ist aktuell noch in Abstimmung und wird unter der VDA-Empfehlungsnummer 4813 erscheinen.

Für die Standardisierung des Austauschs sind mehrere Schritte nötig:

Als erster Schritt wurde festgelegt, die oberste Ebene der Instanzen mit der Firmenkennung (Klasse Enterprise) zu versehen. Dies dient der eindeutigen Zuordnung der Daten zu einer Firma.

Unter diesem Hauptknoten erfolgt die Aufteilung nach Produkt (Stückliste, bill of materials, BOM), Prozess (bill of process, BOP) und Ressourcen (bill of equipment, BOE). Da sich die Ablaufsimulation derzeit ausschließlich aus der Ressourcenliste bedient, wird hier auf eine Detaillierung der anderen beiden Bäume verzichtet.

Bei der Festlegung des Ressourcenbaums wurde beschlossen, sich an die internationale Norm IEC 62264-1 anzulehnen. Abbildung 3 zeigt auf der linken Seite die Einteilung nach IEC, auf der rechten Seite die Ableitung für die VDA-Empfehlung. Die Struktur des Baumes ist sehr schlank gehalten und kann bei Bedarf über Gruppen auf den betreffenden Ebenen ergänzt werden.

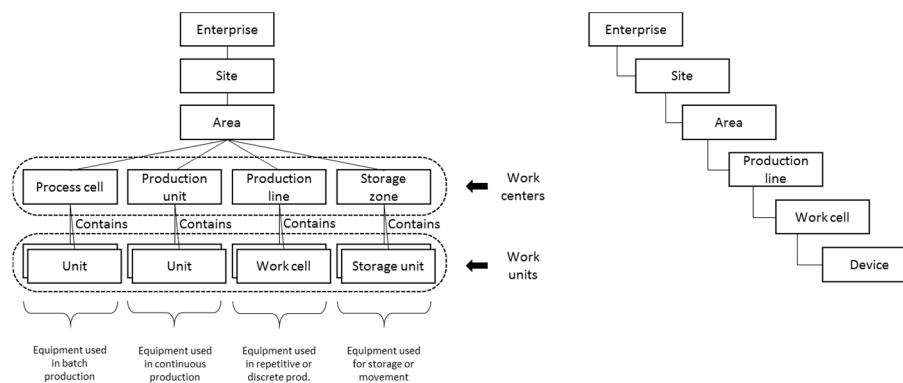


Abbildung 3: Struktur gemäß IEC 62264-1

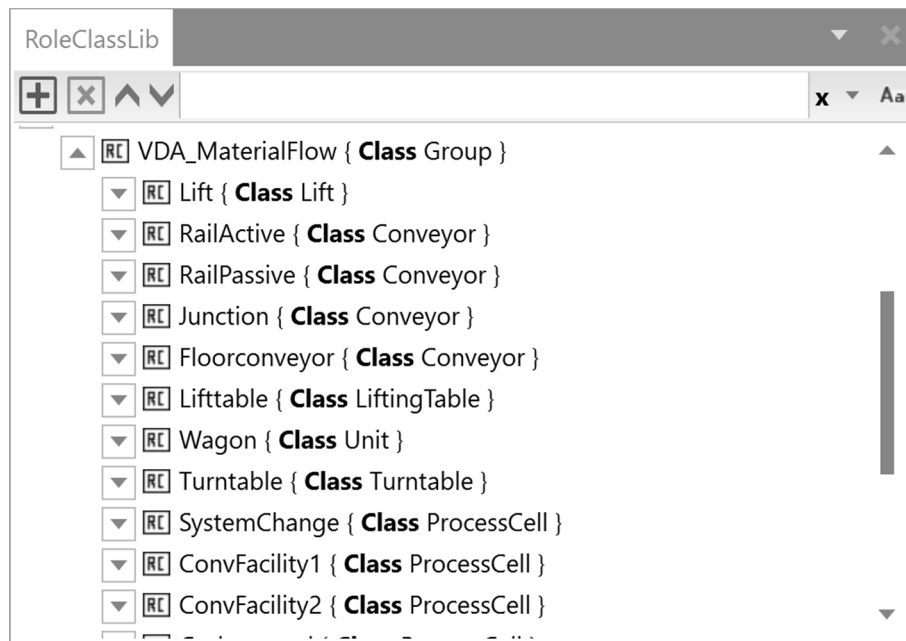
Im nächsten Schritt wurden nun die Bibliotheken (Rollen, Interfaces), die Zuweisung der Attribute (Attributsemantikliste) und der Objekte (Objektsemantikliste) sowie die Interfacesemantikliste definiert. Diese Schritte sind in Abschnitt 4 beschrieben.

Mit Hilfe dieser VDA-Empfehlung soll es ermöglicht werden, beliebige Daten zwischen den Disziplinen der digitalen Fabrik auszutauschen.

#### 4 Umbau der GSL-FT-Schnittstelle

Um die Definitionen der VDA-Empfehlung validieren zu können, wurde innerhalb der VDA-Projektgruppe *AutomationML* beschlossen, dass die festgelegten Punkte anhand von bereits vorhandenen Schnittstellen getestet werden. Dafür wurde u. a. die GSL-FT-Schnittstelle ausgewählt, da bei dieser sowohl abgebende als auch aufnehmende Schnittstellen in der Verantwortung einer VDA-Arbeitsgruppe liegen. Des Weiteren lagen schon im Vorfeld Anfragen von anderen Disziplinen der digitalen Fabrik vor, die die Daten der Fördertechnik nutzen möchten. Ein Beispiel ist die Modellerstellung für die virtuelle Inbetriebnahme (vgl. Abb. 2).

Um die bereits vorhandene proprietäre Schnittstelle migrieren zu können, wurden im ersten Schritt die Klassen der Rollenklassenbibliothek definiert. Dabei wurde versucht, auf vorhandene Bibliotheken des AML-Vereins aufzusetzen (Abb. 4). Da die Objekte in GSL-FT bereits definiert waren, wurde hier eine 1:1-Umsetzung in AML-Rollen gewählt und diese von den AML-Basisklassen abgeleitet.



*Abbildung 4: Die VDA-Rollenklassenbibliothek*

Im nächsten Schritt wurde eine Datei für das Mapping der möglichen Attribute erstellt. Dies ist notwendig, da die Namen der Eigenschaften nicht normiert sind und somit ein und derselbe Wert unterschiedliche Namen in den Systemen haben kann. Das können zum einen einfache Spracheinstellungen sein (z. B. Verfügbarkeit und Availability), zum andern aber generische, nicht selbstsprechende Werte (z. B. Verfügbarkeit und Value\_17). Um diese Problem lösen zu können, wurden eindeutige Bezeichnungen für alle Attribute definiert. Diese Definition wurde an den eCl@ss-Standard angelehnt (vgl. eCl@ss 2017). Als Präfix wurden hier die Zeichenfolge „VDA“ gewählt, um die Attribute eindeutig zuweisen zu können. Ebenso wurde eine Versionierung für Attributnamen eingeführt, die durch die hinteren Zahlen des Namens definiert sind (vgl. Abb. 5, Spalte 1).

Bezeichnung VDA	Beispieldaten	Typ	Einheit	Beschreibung	Attributname GSL-FT
VDA#02-Y000CA#001	Erlang	string		Formbeschreibung für Simulation	mtbf_parameter
VDA#02-Y000CB#001	NegExp	string		Formbeschreibung für Simulation	mttr_parameter
VDA#02-Y000CC#001	1.123	double		X- Koordinate des Startpunkts	Startpoint X
VDA#02-Y000CD#001	1.123	double		Y- Koordinate des Startpunkts	Startpoint Y
VDA#02-Y000CE#001	1.123	double		Z- Koordinate des Startpunkts	Startpoint Z
VDA#02-Y000CF#001	1.123	double		X- Koordinate des Endpunktes	Endpoint X
VDA#02-Y000D0#001	1.123	double		Y- Koordinate des Endpunktes	Endpoint Y
VDA#02-Y000D1#001	1.123	double		Z- Koordinate des Endpunktes	Endpoint Z
VDA#02-Y000D2#001	1.123	double		X- Koordinate des Zentrums	Center X
VDA#02-Y000D3#001	1.123	double		Y- Koordinate des Zentrums	Center Y
VDA#02-Y000D4#001	1.123	double		Z- Koordinate des Zentrums	Center Z
VDA#02-Y000D5#001	;H;-H+;H_V;h;-h+;h_v	string		Art des Bogens	Arctype
VDA#02-Y000D6#001	45	double	Grad	Radius des Bogens	Radius
VDA#02-Y000D7#001		char		Warteposition	Waitpos
VDA#02-Y000D8#001		double			Dog_Space
VDA#02-Y000D9#001		double	m	Stop Abstand	Stop_Dist
VDA#02-Y000DA#001	2	double	m/s	Beschleunigung	Acceleration
VDA#02-Y000DB#001	2	double	m/s	Abbremsung	Deceleration
VDA#02-Y000DC#001	true	boolean		Auffholen möglich	Accumulating
VDA#02-Y000DD#001	2.2	double	m/s	Rotationsgeschwindigkeit	Rotspeed
VDA#02-Y000DE#001	2,3	double	m/s	Geschwindigkeit	Velocity
VDA#02-Y000DF#001	2.3	double	s	Entladezeit	Unloadtime
VDA#02-Y000EG#001	20	double	s	Refresh- Zeit	Refreshtime
VDA#02-Y000E1#001	10	double	s	Transferzeit	Transfertime
VDA#02-Y000E2#001	false	boolean		Lückenziehen möglich	Lueckenziehen
VDA#02-Y000E3#001	0.1	double	m	Abstand zwischen zwei Fahrzeugen	Move_dist
VDA#02-Y000E4#001	4.4	double	m	Länge der beweglichen elemente	MU_Len
VDA#02-Y000E5#001	12	integer		Anzahl der Fahrwerke	MU_Quant

Abbildung 5: Beispiel Attribut-Mapping

Nachdem die Rollen, Klassen und die dazugehörigen Attribute definiert worden sind, mussten noch die Schnittstellen festgelegt werden. Als externe Verknüpfung wurde die AML-Standardverknüpfung für Collada-Dateien gewählt, um die Geometriedaten anzubinden.

Für die internen Vorgänger – Nachfolger existierten zwei unterschiedliche Konzepte, zum einen direkte Verknüpfungen der einzelnen Objekte über Interfaces, zum anderen die Verknüpfung über separate Objekte als Repräsentanz für den Übergabeort. Für die Umsetzung der Fördertechnik wurde die zweite Möglichkeit gewählt, um in späteren Versionen der Schnittstelle die Übergabeorte genauer definieren zu können.

Im finalen Schritt wurden die CAD-Objekte mit ihren Attributen den neu definierten Klassen und Werten zugewiesen. Hierfür wurden zwei einfache Tabellen erstellt, die die Verknüpfung und somit das Schreiben der AML-Datei ermöglichen. Dieses Vorgehen bietet, im Falle einer Änderung an den Klassen, den Vorteil, dass nicht jedes Mal der Exporter neu programmiert werden muss, sondern nur die Tabellen aktualisiert werden müssen. Die gleichen Tabellen wurden dann noch für die Importer definiert. Somit ist ein Datenaustausch übergreifend mit AML möglich.

Um die Schnittstelle umzusetzen, wurde im ersten Schritt ein Konverter programmiert, der die GSL-FT-Datei in AML umsetzt. Dieser Schritt war nötig, um zu testen, ob die definierten Rollen und Klassen sowie das Format selbst valide sind. Der zweite Schritt war dann die Umsetzung in der abgebenden Software, in diesem Fall TriCAD. Dort ist es nun möglich, die Fördertechnikdaten entweder im GSL-FT-Format oder aber in AML zu exportieren.

Die erste Erweiterung der Daten ist durch den Export der Geometriedaten geschehen. In AML werden für alle im Ressourcenbaum befindlichen Objekte auch die Collada-Dateien erzeugt und in der AML-Datei referenziert. Somit wäre auch der Aufbau eines

dreidimensionalen Simulationsmodells möglich. Dies wird jedoch aktuell noch nicht genutzt, da nicht jede Simulationssoftware Collada nativ lesen kann und die Daten somit noch gewandelt werden müssten.

Zum Schluss mussten dann noch die aufnehmenden Systeme angepasst werden. Dies wurde zuerst für den VDA Automotive Bausteinkasten für Plant Simulation sowie für die VIBN-Software WinMod/Simline umgesetzt und steht Ende 2017 zur Verfügung. Des Weiteren ist es möglich, die Geometriedaten für eine bewegte 3D-Visualisierung (Interviews3D) zu nutzen, wobei hier die Geometrien aus der AML-Datei genommen werden und die Bewegungen aus dem Simulationswerkzeug berechnet und in eine Collada-Datei ausgeleitet werden.

## 5 Ausblick

AML ist ein mächtiges Werkzeug, um Daten für die digitale Fabrik neutral auszutauschen. Das gilt nicht nur für die Automobilindustrie, sondern für jegliche automatisierte Produktion. Erste Einsätze zeigen bereits, wie vielschichtig der Einsatz ist. Aber Vorsicht ist geboten. Ohne weitere branchenspezifische Normierungen wird ein Datenaustausch auch in Zukunft nicht immer ohne Konverter möglich sein.

Übergreifend definierten Klassen und Attributen, inklusive dazugehörigem Mapping für jede einzelne Software, haben sich als gute Lösung etabliert und kommen so auch weit über die automobilen Einsatzgebiete hinaus zum Einsatz.

Als nächste Schritte werden die Werkzeuge der Automatisierungstechnik angebunden. So wurde z. B. bereits die Elektroplanungssoftware EPlan um eine AML-Schnittstelle erweitert. Bei diesen Anbindungen wird sich auch herausstellen, ob die Inhalte der aktuellen Definition ausreichen oder ob diese erweitert werden müssen.

Ein noch nicht gelöstes Problem sind die Grafikformate. Hier konkurriert das in AML implementierte Format Collada mit den in den meisten Product Lifecycle Management Systemen vorhandenen Formaten (z. B. JT, Jupiter Tessellation). Es ist aber bereits ein Lösungsvorschlag beim AML-Verein beantragt und wird aktuell geprüft.

## Literatur

AutomationML: [www.automationml.org](http://www.automationml.org). Letzter Zugriff am 06.07.2017.

Collada: [www.khronos.org/collada/](http://www.khronos.org/collada/). Letzter Zugriff am 06.07.2017.

Drath R.: Datenaustausch in der Anlagenplanung mit AutomationML. Berlin: Springer 2010.

eCl@ss: [www.eclass.eu/](http://www.eclass.eu/). Letzter Zugriff am 06.07.2017.

Jörg, T.; Wacker, R.: Generische Simulationslösung Fördertechnik. In: Rabe, M. (Hrsg.): Advances in Simulation for Production and Logistics Applications. Stuttgart: Fraunhofer-IRB 2008, S. 11-20.

PLCopen: [www.plcopen.org](http://www.plcopen.org). Letzter Zugriff am 06.07.2017.

Rabe, M.; Spieckermann, S.; Wenzel, S.: Verifikation und Validierung für die Simulation in Produktion und Logistik. Berlin, Heidelberg: Springer 2008.