

# **Auf dem Weg zum Standard – Virtuelle Inbetriebnahme von IT-Steuerungssystemen in der Produktionssteuerung**

## ***The Way to Standardization – Virtual Implementation of IT Control Systems in Production Control***

Ulrich Grillitsch  
BMW AG, Regensburg (Germany)  
ulrich.grillitsch@bmw.de

Gottfried Mayer  
BMW AG, München (Germany)  
gottfried.mayer@bmw.de

**Abstract:** After finishing some virtual ramp up's in separate plants and subsections like bodyshop, paintshop and motor compartment using emulation, there is the need for a standardization and improvement of emulation applications. This article shows a standardization approach using the example of a specific project.

## **1 Einleitung**

Nachdem die reale Inbetriebnahme der Produktionssteuerungssysteme in verschiedenen Werken und Fertigungsabschnitten in den Technologien Karosseriebau, Lackiererei und Motorenbau mittels Emulation (FOLLERT, TRAUTMANN 2006, S. 521 ff.) erfolgreich verlaufen ist (MAYER, BURGES 2006, S. 541 ff.), stellt sich die Frage nach einer verbesserten Vorgehensweise, die mit einfachen, schlanken sowie parametrisierbaren Bausteinen ein schnelles Aufbauen von Emulationsmodellen ermöglicht, die in einem breiten Anwendungsgebiet eingesetzt werden können.

Die Potentiale einer Verbesserung der Vorgehensweise sind:

- Deutliche Reduzierung des Modellierungsaufwandes.
- Vereinfachung von Betrieb und Wartung der Modelle.
- Unabhängigkeit vom verwendeten Steuerungssystem.
- Erleichterung von Know-How-Transfer sowie Wissenssicherung.

- Deutliche Verkürzung der Inbetriebnahmezeiten und der Aufwände in den beteiligten Fachbereichen und IT-Stellen.
- Ausweitung der Nutzung für Schulungen von Steuerungs- und Instandhaltungspersonal.

Gegebene Randbedingung für die Standardisierung ist die Nutzung des VDA (Verband der Automobilindustrie) Bausteinkastens (MAYER, HEINRICH 2006, S. 423 ff.).

Dieser Artikel ist wie folgt strukturiert: im zweiten Kapitel wird die Vorgehensweise zur Standardisierung beschrieben. Das dritte Kapitel beinhaltet eine Projektbeschreibung zur Verdeutlichung der im vorherigen Kapitel beschriebenen Maßnahmen. Im vierten Kapitel ist eine Zusammenfassung sowie einen Ausblick auf weitere Vorhaben im Bereich der virtuellen Inbetriebnahme zu finden.

## 2 Vorgehensweise

Aus einer Analyse der bislang angewendeten Bausteine und des Ablaufs vergangener Projekte ergeben sich eine Reihe von Ansätzen zur Standardisierung und Verbesserung. Beides, die Analyse und das Ergebnis von Verbesserungen, werden in diesem Kapitel beschrieben.

### **Analyse der vorhandenen Modelle und Bausteine:**

Bei der Analyse hat sich herausgestellt, dass die für den Telegrammverkehr benötigten Informationen in den Simulationsmodellen sehr schwer zu finden und in ihrer Bedeutung kaum nachvollziehbar sind. So wurden Telegramminhalte zum Teil in Methoden als feste Zeichenketten abgespeichert sowie in verschiedenen Tabellen an unterschiedlichen Stellen im Modell abgelegt. Ferner wurden Tabellen nicht zentral gepflegt, sondern mussten in den einzelnen Instanzen bearbeitet werden. Dadurch wurde der Aufwand für den Modellaufbau erheblich höher und auch fehleranfälliger. Die vorhandenen Emulationsmodelle hatten unterschiedliche Bausteinstände, die kaum konsolidierbar waren. Als Folge war eine Wartung der Modelle kaum noch möglich. Bei der Modellierung wurde nicht durchgängig auf die Vererbungshierarchie geachtet, somit waren gleiche Bausteine in unterschiedlichen Ordnern abgelegt und abgeleitet. Die Updatefähigkeit, die einer der großen Vorteile bei der Verwendung des VDA-Bausteinkastens ist, war damit praktisch nicht mehr gegeben.

Eine weitere Besonderheit ergibt sich auch aus der Socket-Schnittstelle, die Grundlage der Kommunikation zwischen Simulationsmodell und realem IT-System ist. Die in der Socket-Schnittstelle des Simulators Plant Simulation implementierte TCP-IP Protokollversion entspricht nicht dem RFC1006 Standard und bildet bei Lasttests Telegrammpakete, welche die reale Steuerung nicht verarbeiten kann. Im Einzelnen gibt es die folgenden Unterschiede:

### **Verhalten des TCP Protokolls:**

Wenn Daten mit Hilfe des TCP Protokolls übertragen werden, dann erfolgt die Übertragung datenstromorientiert. Es werden dabei weder Informationen zur Länge noch Informationen über Anfang und Ende einer Nachricht übertragen. Beim Senden ist dies unproblematisch, da der Sender weiß, wie viele Datenbytes er ver-

schicken will. Der Empfänger kann jedoch nicht erkennen, wo eine Nachricht im Datenstrom aufhört und wo die nächste im Datenstrom beginnt.

#### ***Verhalten der RFC1006 Protokollerweiterung:***

In den meisten Anwendungen in der Automatisierungstechnik ist es jedoch unerlässlich, nachrichtenorientiert zu arbeiten. Es werden abgeschlossene Nachrichtenblöcke über eine Verbindung gesendet, die vom Empfänger auch als solche erkannt werden. Die Protokollerweiterung RFC1006 spezifiziert nun, welche Informationen in Form eines als Header bezeichneten Telegrammkopfs den zu übertragenden Daten hinzugefügt werden müssen, um dies zu gewährleisten. RFC1006 stellt also Anwendungen, die auf dem datenstromorientierten TCP Protokoll aufsetzen, eine nachrichtenorientierte Übertragung zur Verfügung.

#### **Neustrukturieren und Vereinfachen der Bausteine:**

Auf Basis der Analyseergebnisse wurden die Bausteine überarbeitet und neu entwickelt. Die Anzahl der Objekte hat sich dabei erheblich reduziert, wodurch die Übersichtlichkeit deutlich erhöht wurde. Im Wesentlichen wurden die einzelnen Telegrammtabellen in einer zentralen Tabelle zusammengefasst und in Ihrer Funktionalität erweitert. Die für jede Telegrammtabelle existierende Verarbeitungsmethode wurde entfernt und deren Funktionsweise in einer zentralen Methode integriert und dynamisiert. Es wurde darauf geachtet, dass keine hartcodierten Zeichenketten in den Methoden verwendet werden.

Die speziellen Zusatzanforderungen einer sogenannten Master-SPS (Speicherprogrammierbare Steuerung) wurden in den Standardtelegrammablauf integriert. Master-SPSen können unabhängig vom Materialfluß Telegramme senden, z. B. Auftragsanforderungen. Ebenso muß der Master-SPS Baustein in diesem Standardablauf Telegramme verarbeiten können, die direkt von der übergeordneten Steuerung gesendet werden. Ein Emulationsmodell kann viele SPS- und Materialflußbausteine enthalten. Um nicht jedes Objekt einzeln aufrufen und parametrisieren zu müssen, wurde die Konfiguration der Bausteine in einer zentralen Tabelle zusammengeführt. Das hat auch den Vorteil, dass beim Aktualisieren der Bausteine die Parametrisierungen einfacher gesichert und wieder aufgeprägt werden können (siehe auch Kapitel 3, Randbedingungen).

#### **Einrichten eines TCP-Protokollkonverters:**

Der intern entwickelte Protokollkonverter übernimmt die oben beschriebene, in der Socket-Schnittstelle von Plant Simulation fehlende Funktion des TCP-IP RFC1006 Protokolls. Er wird als Prozeß auf der Kommunikationsmaschine der übergeordneten Steuerung installiert.

#### **Integration von An- bzw. Abschloptionen einzelner Bausteinkomponenten:**

Die Flexibilität und Nutzbarkeit von Modellen soll beim Einsatz so hoch wie möglich sein. Am Beispiel des sogenannten "OFFLINE-Telegrammtests" wird der Punkt erläutert: Um ein Modell auch ohne Verbindung zur realen Steuerung aufbauen und testen zu können, wurde eine Möglichkeit geschaffen, Antworttelegramme im Modell selbst zu erzeugen und damit eine Verbindung zur Steuerung zu simulieren. Dadurch kann zum einen getestet werden, ob die Telegramme richtig parametrisiert wurden und zum anderen, ob die internen Modellsteuerungen funktionieren. Das

Modell hat dadurch zum Zeitpunkt der ersten Funktionstests mit der realen Steuerung bereits einen ausgereiften Stand.

**Erstellen von verbindlichen Referenzobjekten:**

Alle für die Emulation relevanten Objekte existieren in einem einzigen verbindlichen Objektstand. Alle Änderungen in bestehenden Modellen werden in das Referenzobjekt übernommen. Dadurch ist gewährleistet, dass alle Emulationsmodelle mit den gleichen Bausteinen aufgebaut werden und mit geringem Aufwand aktualisiert werden können. Damit fließt auch automatisch die Erweiterung der Funktionalität in alle Modelle ein.

**Anlegen einer klaren und durchgängigen Vererbungshierarchie:**

Dieser Schritt ist zwingend notwendig, um die oben bereits erwähnte Updatefähigkeit der Bausteine zu gewährleisten. So kann durch die Export- und Importfunktion von Plant Simulation das komplette Emulationsobjekt ausgetauscht werden. Um dabei nicht die Parametrisierungen der Instanzen zu verlieren, wurde die Möglichkeit geschaffen, auf Knopfdruck alle Parameter auszulesen und in einer zentralen Tabelle abzuspeichern. Nach erfolgreichem Import können wiederum auf Knopfdruck alle Parameter auf die Instanzen (Bausteine) aufgeprägt werden.

**Dialogsteuerung für jedes Emulationsmodell:**

Da im Regelfall die Modelle auch vom Inbetriebnahmepersonal verwendet werden, das in der Handhabung des Simulationswerkzeugs nicht geschult ist, muß eine einfache Steuerung des Modells gewährleistet sein. Abgesehen vom Modellstart und -stop, Verbindungsaufbau und -abbau können über Anwenderdialoge Auftragslisten erstellt und bearbeitet, das SPS-Verhalten sowie das Materialfluß-Verhalten beeinflusst werden. Auch für den Simulations- bzw. Emulationsexperten bedeutet die Dialogsteuerung eine Erleichterung bei der Bedienung.

**Durchgängige und verständliche Dokumentation:**

Für jedes einzelne Modell wird ein Benutzerhandbuch für das Inbetriebnahmepersonal erstellt. Für die Modellentwickler wird laufend ein Emulationshandbuch zur Wissenssicherung auf den neuesten Stand gebracht.

**Abstraktion des Materialflusses:**

Die Priorität liegt bei der virtuellen Inbetriebnahme der übergeordneten Steuerungssoftware. Daher wird im Bereich der Fördertechnik abstrahiert, um unnötige Modellierungsaufwände zu vermeiden. Je nach Anforderung werden auch logische Layouts, wie sie in der Realität für Visualisierungen der Steuerungssoftware verwendet werden, eingesetzt.

Einen Überblick über den Telegrammverkehr innerhalb eines Emulationsmodells bei Verwendung der überarbeiteten Bausteine gibt Abb. 1.

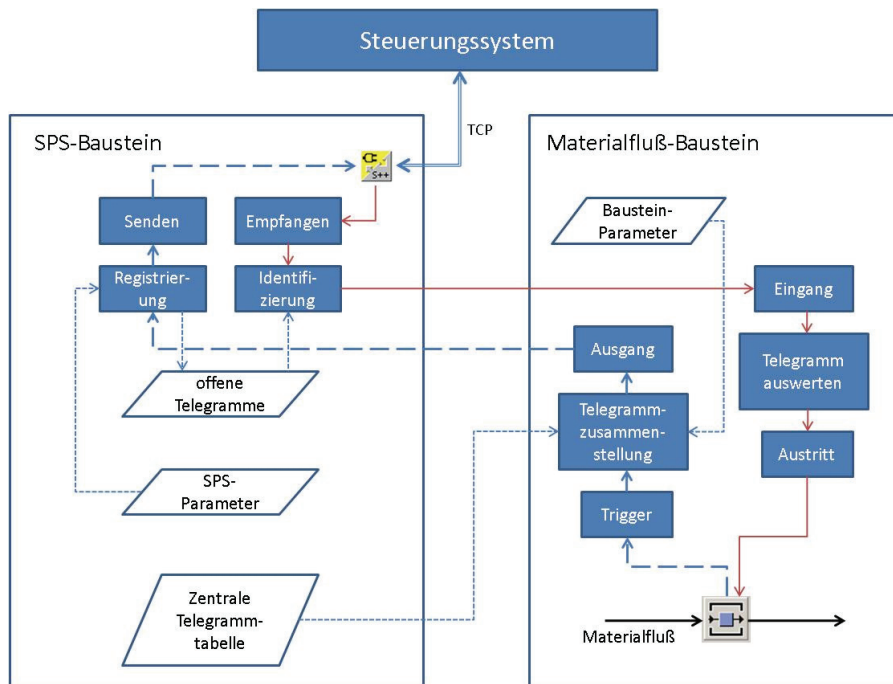


Abbildung 1: Schematischer Ablauf des Telegrammverkehrs

### 3 Projekt Kabelbaum-Lagersteuerung

Eine erste Validierung der standardisierten Bausteine erfolgte im Rahmen eines Arbeitsauftrages zur Unterstützung der Umstellung des Kabelbaumlagers im Werk Dingolfing auf die Lagersteuerungssoftware ASL.

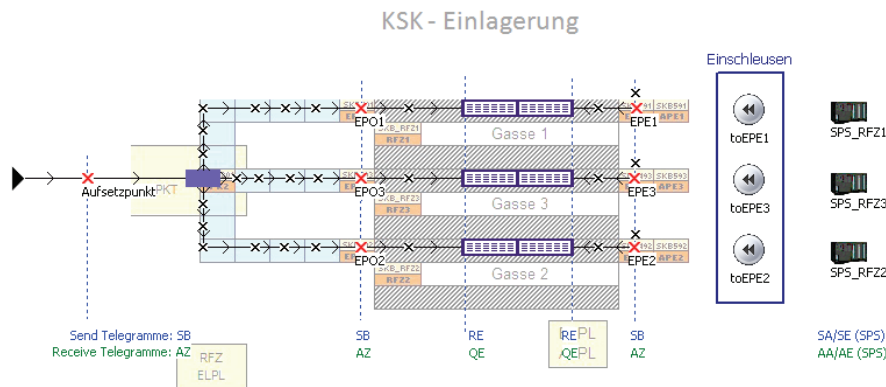
Im Einzelnen wurden die folgenden Randbedingungen für die Implementierung festgelegt:

- Keine Vorbelegung der Fördertechnik im Simulationsmodell, um Inkonsistenzen zwischen Modell und Steuerungs-Datenbank zu vermeiden.
- Vernichtung der Behälter im Modell nach erfolgter Einlagerung. Dadurch entfällt eine aufwendige, modellinterne Speicherverwaltung, die nicht Fokus der Emulation ist.
- Erzeugen der Behälter beim Empfang von Auslageraufträgen.
- Leerfahren des Systems nach jedem Funktions- und Lasttest.
- Änderung der angezeigten Behältersymbole durch eine Bildsteuerung. So können Behälter, die sich gerade im Telegrammverkehr befinden, identifiziert werden (vgl. Abbildung 3).
- Einstellmöglichkeit, welche Quellen Behälter erzeugen sollen und welche nicht. Interessant ist die Aktivierung bzw. Deaktivierung einzelner Quellen, wenn nur Modellabschnitte bzw. Teilfunktionen getestet werden sollen.

- Zentrale Verwaltung von Telegrammen in einer Tabelle. In dieser Tabelle wird pro Telegrammname die Zuordnung zu den Materialflußbausteinen, die dieses Telegramm senden, sowie deren Struktur eingestellt (vgl. Abbildung 1, SPS-Baustein).
- Zentrale Verwaltung von den Materialflußbausteinen mit Telegrammverkehr in einer Tabelle. In dieser Tabelle werden pro Materialflußbaustein alle relevanten Parameter, die für den Telegrammverkehr notwendig sind, eingestellt (vgl. Abbildung 1, Materialfluß-Baustein).
- Dynamische, tabellengesteuerte Master-SPS Funktion (zyklische SPS-Telegramme, die nicht von einem Behälter im Materialfluß ausgelöst werden).

Getestet worden sind in diesem Projekt der Einlagerungs- und der Auslagerungsprozess. Die entsprechenden Abläufe werden im Folgenden kurz skizziert.

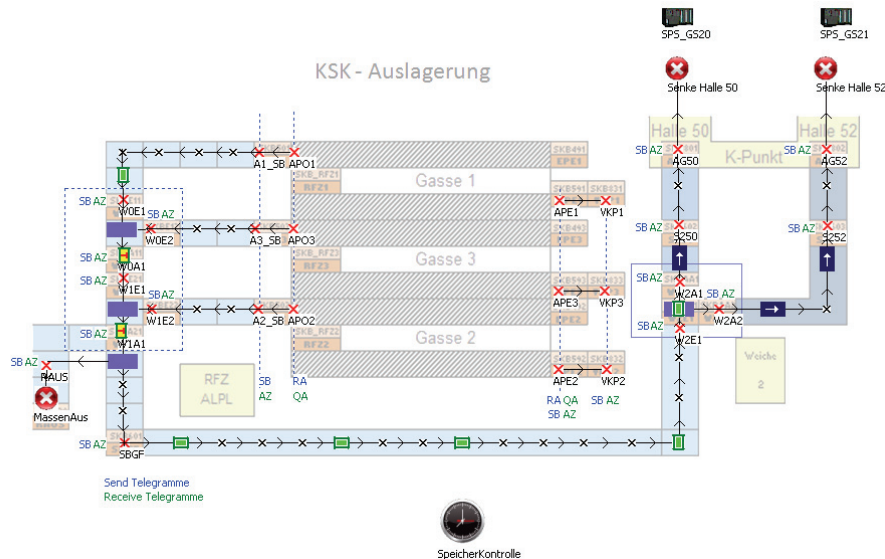
Zu Beginn des Einlagerungsprozesses (vgl. Abbildung 2) werden die Kabelbäume am Aufsetzpunkt mit einem Telegramm von der ASL Steuerung erfasst und in eine der drei Gassen disponiert. Wenn sich ein Kabelbaum am Gasseneingang meldet, wird er vom Regalfahrzeug übernommen und eingelagert. Voraussetzung hierfür ist jedoch, dass ein Einlagerungsauftrag von ASL an die SPS des jeweiligen Regalfahrzeugs (RFZ) einer Gasse geschickt und von der SPS empfangen wurde. Die SPS sendet solange zyklisch Einlagerungsanforderungen an ASL, bis eine Einlagerung ausgeführt wird. Erst wenn die Einlagerung mit einem Telegramm von ASL quittiert wurde, wird wieder die zyklische Einlagerungsanforderung von der SPS gesendet.



**Abbildung 2:** Layout Einlagerung Kundenspezifischer Kabelbaum (KSK)

Im Rahmen des Auslagerungsprozesses (vgl. Abbildung 3) senden die SPSen der drei Gassen zyklisch Auslagerungsanforderungen an ASL. Wird von ASL ein Auslagerungsauftrag an die Regalfahrzeug-SPS einer Gasse gesendet, wird von der Speicherkontrolle ein Kabelbaum-Behälter erzeugt, mit den relevanten Telegramminformationen versehen, auf den Auslagerplatz umgelagert und ein Auslagervollzugs-Telegramm gesendet, das von ASL quittiert wird. Erst danach sendet die RFZ-SPS erneut eine Auslagerungsanforderung an die übergeordnete Steuerung (ASL). Jetzt befindet sich der Kabelbaum auf der Förderstrecke und sendet an den Dispo-Punkten Telegramme, hauptsächlich um dem Kabelbaum eine Zielvorgabe mitzugeben.

Nach dem Modellaufbau wurden zuerst Funktionstests mit einzelnen Kabelbaum-Behältern durchgeführt. Bereits in dieser Phase konnten Fehler sowohl im Emulationsmodell als auch in der Steuerungssoftware entdeckt und behoben sowie Verbesserungen implementiert werden. Anschließend erfolgten umfangreiche Lasttests, die mit verschiedenen ASL Konfigurationen durchgeführt wurden. Nach dem Abschluss der Tests konnte die reale ASL Lagersteuerung fehlerfrei in Betrieb genommen werden.



**Abbildung 3:** Layout Auslagerung kundenspezifischer Kabelbaum (KSK)

Wäre die virtuelle Inbetriebnahme des Kabelbaumlagers auf Basis der nicht standardisierten Bausteine erfolgt, hätte das folgende Auswirkungen auf das Projekt gehabt:

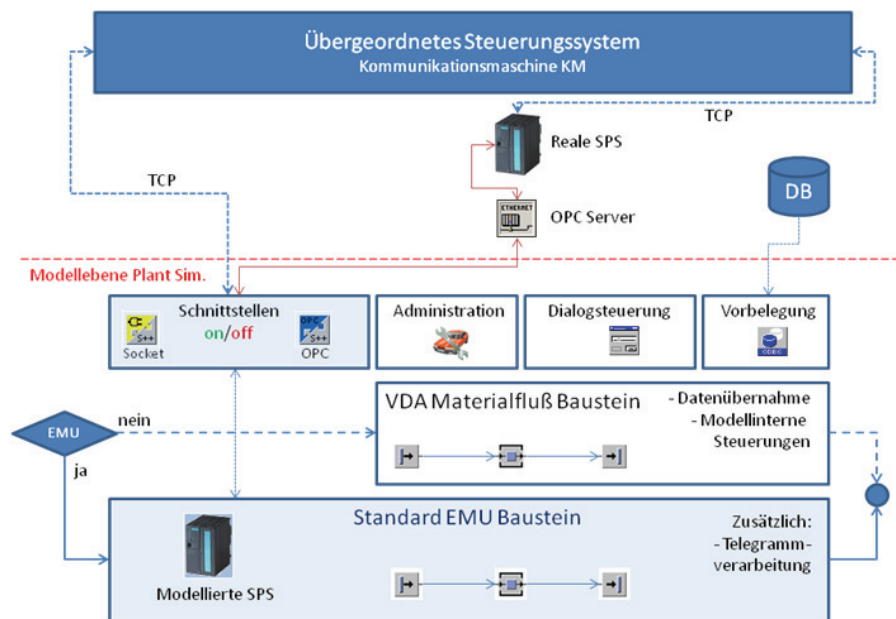
- Gefährdung der vorgegebenen Inbetriebnahme-Termine.
- Erheblicher Mehraufwand beim Modellaufbau.
- Analyse-Probleme bei Funktions- und Lasttest-Fehlern.
- Eingeschränkte Wiederverwendbarkeit des Modells für nachfolgende, ähnlich aufgebaute Lager-Inbetriebnahmen.

#### 4 Zusammenfassung und Ausblick

Durch Umsetzung der in diesem Artikel beschriebenen Maßnahmen zur Standardisierung von Emulationsbausteinen konnte eine deutliche Reduzierung der Modellierungs- und Inbetriebnahme-Aufwände erzielt werden. Im vorgestellten Beispiel hat das maßgeblich dazu beigetragen, dass die reale Inbetriebnahme mit Hilfe der Emulation erfolgreich und termingerecht abgeschlossen werden konnte. Der Weg der Standardisierung der Emulationsbausteine hat sich bewährt und wird kontinuierlich weiter verfolgt. Unabhängig von der durchzuführenden Inbetrieb-

nahme läßt sich prognostizieren, dass unter Beachtung der aufgezeigten Schritte und Maßnahmen auch in anderen Fertigungsbereichen ähnliche Ergebnisse und Aufwandsreduzierungen erreichbar sind.

Nicht nur die standardisierten Kommunikationsbausteine der Emulation sind ein wesentlicher Bestandteil einer effizienten virtuellen Inbetriebnahme, sondern auch der Zugriff auf standardisierte Simulationsmodelle im Unternehmen, die nach einer Struktur aufgebaut sind, die die Integration dieser Kommunikationsbausteine ohne Probleme ermöglichen (vgl. Abbildung 4). Dazu ist es notwendig, schon bei der Erstellung von sogenannten Betreiber- und Planungsmodellen die Anforderungen der Emulation zu kennen, zu berücksichtigen und umzusetzen. Dieser Aspekt wird derzeit konzeptionell ausgearbeitet und danach in einer Pilotanwendung auf Praktikabilität geprüft.



**Abbildung 4:** Schema eines kombinierten Simulations-/Emulationsmodells

## Literatur

- FOLLERT, Guido; TRAUTMANN, Andreas: Emulation intralogistischer Systeme. In: Simulation in Produktion und Logistik 2006. Hrsg.: WENZEL, Sigrid. Erlangen: SCS Publishing House, 2006, S. 521-530.
- MAYER, Gottfried; BURGESS, Ulrich: Virtuelle Inbetriebnahme von Produktionssteuerungssystemen in der Automobilindustrie mittels Emulation. In: Simulation in Produktion und Logistik 2006. Hrsg.: WENZEL, Sigrid. Erlangen: SCS Publishing House, 2006, S. 541-550.
- MAYER, Gottfried; HEINRICH, Stefan: Ablaufsimulation im VDA – ein Bericht aus der Arbeitsgruppe. In: Simulation in Produktion und Logistik 2006. Hrsg.: WENZEL, Sigrid. Erlangen: SCS Publishing House, 2006, S. 423-428.