

Simulationsbasierte Frühwarnsysteme zur Unterstützung der operativen Produktionssteuerung und -planung in der Automobilindustrie

Dissertation

zur Erlangung des akademischen Grades

Doktoringenieur (Dr.-Ing.)

angenommen durch die Fakultät für Informatik
der Otto-von-Guericke-Universität Magdeburg

von: Diplom-Wirtschaftsingenieur Ingo Hotz
geb. am 02. Juni 1978 in Wertheim am Main

Gutachter:

Prof. Dr.-Ing. habil. Thomas Schulze
Prof. Dr. rer. nat. Oliver Rose
Prof. Dr.-Ing. habil. Peter Gmilkowsky

Ort und Datum des Promotionskolloquiums: Magdeburg, den 28. September 2007

Ingo Hotz
*Simulationsbasierte Frühwarnsysteme zur Unterstützung
der operativen Produktionssteuerung und -planung
in der Automobilindustrie*
Dissertation, Otto-von-Guericke Universität
Magdeburg, 2007

Vorwort

Die Ergebnisse dieser Arbeit resultieren aus meiner dreijährigen Tätigkeit als Doktorand bei der DaimlerChrysler AG im Planungsbereich des Werkes Gaggenau und Getriebewerkes Rastatt. Dort beschäftigte ich mich mit den unterschiedlichsten Fragestellungen und Anwendungen rund um das Themengebiet der Materialflusssimulation. Zu meinen Aufgaben gehörte die Durchführung von Simulationsstudien sowohl in der Planungs- und Projektierungsphase als auch in der Betriebsphase, die Anbindung der Simulation an reale Produktionssysteme und die Unterstützung der Einführung Digitaler Planungsmethoden im Nutzfahrzeugbereich. An dieser Stelle möchte ich allen Produktionsplanern, Teamleitern und Meistern danken, die durch ihre Unterstützung zum Gelingen dieser Arbeit beigetragen haben.

Ganz außerordentlicher Dank gilt meinem Doktorvater, Herrn Prof. Dr. Thomas Schulze, der mir mit seinen Anregungen, Vorschlägen und Ideen, sowie seiner Fähigkeit, komplexe Sachverhalte differenziert zu betrachten, die Ausarbeitung meines Themas erst ermöglichte.

Weiterhin möchte ich meinen Eltern danken, die mir zu jeder Zeit die notwendige Unterstützung haben zukommen lassen und mir durch ihre Verlässlichkeit und Beständigkeit immer ein Rückhalt waren.

Besonderen Dank schulde ich meiner Freundin Kerstin, die mich schon seit Jahren durch die entbehrungsreichen Phasen meiner Ausbildung begleitet. Sie gab mir immer den notwendigen Halt und Mut, auch in Zeiten der Stagnation meiner Arbeit, nicht aufzugeben und durchzuhalten.

Patience and diligence, like faith, remove mountains.

— William Penn

Magdeburg, 28. September 2007

Ingo Hotz

Ich erkläre hiermit, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Insbesondere habe ich nicht die Hilfe eines kommerziellen Promotionsberaters in Anspruch genommen. Dritte haben von mir weder unmittelbar noch mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen.

Die Arbeit wurde bisher weder im Inland noch im Ausland in gleicher oder ähnlicher Form als Dissertation eingereicht und ist als Ganzes auch noch nicht veröffentlicht.

Magdeburg, 28. September 2007

Ingo Hotz

Kurzfassung

Die Automobilindustrie ist von einer Globalisierung der Märkte geprägt. Dies bewirkt eine zunehmende Wettbewerbsintensität, Transparenz der Märkte und weltweite Überkapazitäten. Die Automobilkonzerne reagieren auf diese Herausforderungen mit kürzeren Produktlebenszyklen, verstärkter Kundenorientierung, einer Individualisierung von Produktion und Leistung und somit einem größeren Produktportfolio. Eine große Variantenvielfalt und die verstärkte Kundenorientierung bewirken eine steigende Komplexität und Dynamik in der Produktionssteuerung und -planung. Dem dadurch steigenden Kostendruck soll mit möglichst geringen Beständen bei kurzen Durchlaufzeiten und einer höheren Flexibilität der Materialflusssysteme entgegen gewirkt werden. Diese Forderungen setzen eine kurzfristige und schnelle Reaktionsmöglichkeit der Systeme voraus, da sich Störgrößen sehr viel stärker auf die Prozesssicherheit des Materialflusssystems auswirken. Die Vorhersage von Ereignissen, Zuständen oder Entwicklungen ermöglicht einen zeitlichen Vorteil, da frühzeitig geeignete Reaktionen oder Gegenmaßnahmen erfolgen können. Aufgrund der Komplexität realer Materialflusssysteme ist die Simulation besonders zur Prognose künftiger Systemzustände geeignet. Die Einführung Simulationsbasierter Frühwarnsysteme unterstützt die proaktive Steuerung von Materialflusssystemen. Dabei wird ein zeitlicher Vorteil sowohl durch die frühzeitige Erkennung von Ausnahmesituationen als auch durch die Selektion und Bewertung möglicher proaktiver Steuerungsmaßnahmen erreicht.

Es wird eine allgemeingültige Architektur Simulationsbasierter Frühwarnsysteme vorgestellt, welche deren Funktionalitäten unabhängig vom realen System, den verwendeten Simulatoren, der Modellierungsphilosophie und der technischen Implementierung umfassend beschreibt. Simulationsmodelle werden mit dem gegenwärtigen Systemzustand des realen Systems initialisiert, wobei die Informationssysteme der realen Systeme einzubinden sind. Dabei stellt die Anwendung der Simulation zur Prognose künftiger Systemzustände im Umfeld der operativen Produktionssteuerung und -planung besondere Anforderungen an die verwendeten Simulatoren und die Entwicklung von Simulationsmodellen. Die Detektion möglicher Ausnahmefälle wie beispielsweise zu hohe Bestände, Terminabweichungen oder Kapazitätsengpässe spielt dabei eine große Rolle. Werden Ausnahmesituationen in einem Materialflusssystem festgestellt, so kann der zeitliche Vorteil dieser Früherkennung durch eine effiziente Lokalisierung geeigneter proaktiver Steuerungsmaßnahmen noch verstärkt werden. In der Praxis bereits erfolgreich eingesetzte Monitoringsysteme verwenden vordefinierte Handlungsalternativen für vordefinierte Ausnahmesituationen. Im Rahmen dieser Arbeit stellt sich die Frage, ob nicht ein umfassender Ansatz existiert, der auch neue, noch undefinierte Alternativen liefern kann, die auf die Art und den Umfang der Ausnahmesituation abgestimmt sind. Simulationsbasierte Frühwarnsysteme müssen mit einer möglichst effizienten, einfachen und offenen Implementierung unabhängig von einer unternehmensspezifischen IT-Infrastruktur realisiert werden.

Inhaltsverzeichnis

Abbildungsverzeichnis	iv
Tabellenverzeichnis	vi
1 Einführung	1
1.1 Motivation	1
1.2 Fragestellung und Zielsetzung	3
1.3 Aufbau der Arbeit	4
2 Simulation in Produktion und Logistik	6
2.1 Computersimulation	7
2.1.1 Definitionen zur Simulation	8
2.1.2 Klassifikation und Methoden der Simulation	10
2.1.2.1 Diskrete Simulation	11
2.1.3 Anwendungsbereiche, Nutzen und Grenzen	12
2.2 Simulation in der Automobilindustrie	14
2.2.1 Informationstechnische Integration der Simulation	15
2.2.1.1 Computer Integrated Manufacturing	16
2.2.1.2 Produktionsplanungs- und Steuerungssysteme	16
2.2.1.3 Die Betriebsdatenerfassung	17
2.2.2 Die Digitale Fabrik	18
2.2.3 Simulation in Produktion und Logistik	19
2.2.3.1 Simulationswerkzeuge	21
2.2.3.2 Vorgehensweise bei der Modellerstellung	23
2.3 Entwicklungstendenzen der Simulation in Produktion und Logistik	24
2.4 Zusammenfassung des Kapitels	26
3 Simulationsbasierte Frühwarnsysteme	27
3.1 Frühwarnsysteme	27
3.2 Simulationsbasierte Frühwarnsysteme in Produktion und Logistik	30
3.2.1 Eine Definition	31
3.2.2 Anforderungen an Simulationsbasierte Frühwarnsysteme	33
3.2.3 Abgrenzung der Problemstellung	35
3.2.3.1 Hardware-In-The-Loop- und Real-Time-Simulation	36
3.2.3.2 Betreibersimulation	37
3.2.3.3 Leitstandssimulation	37
3.2.3.4 Einordnung Simulationsbasierter Frühwarnsysteme	38
3.3 Eine allgemeine Architektur Simulationsbasierter Frühwarnsysteme	39

3.3.1	Hauptkomponenten eines SEWS	39
3.3.2	Der Aufbau des Frameworks	41
3.4	Verteilte Systeme und Kommunikationsprinzipien	44
3.4.1	RPC und RMI	45
3.4.2	CORBA und DCOM	45
3.4.3	Web Services	46
3.4.4	Eingesetztes Kommunikationsprinzip	47
3.5	Steuerung des Workflows	49
3.5.1	Interaktion der Komponenten eines SEWS	50
3.5.2	Möglichkeiten zur Steuerung des Workflows	51
3.5.2.1	Spezifische Anwendungslösung	52
3.5.2.2	Die Business Process Execution Language	53
3.5.2.3	Zyklische Durchführung der Abläufe	55
3.6	Zusammenfassung des Kapitels	55
4	Prognose künftiger Zustände des realen Systems	57
4.1	Datenbeschaffung und -verwaltung	57
4.1.1	Simulations- und SEWS-relevante Daten	58
4.1.2	Kriterien zur Systemauswahl	61
4.1.3	SEWS-interne Datenverwaltung mit XML	63
4.1.4	Verwaltung mit Hilfe einer XML-Datenbank	66
4.1.5	Datengewinnung aus originären Datenquellen	67
4.1.6	Aktualisierung von Modelldaten	70
4.2	Simulation als eingebettete Applikation	71
4.2.1	Relevante Eigenschaften von Simulatoren	72
4.2.2	Anforderungen an Simulationsmodelle	75
4.3	Simulation zur Vorhersage künftiger Zustände des realen Systems	77
4.3.1	Initialisierung von Simulationsmodellen	77
4.3.1.1	Datenübertragung und -bereitstellung	78
4.3.1.2	Initialisierung	79
4.3.2	Prognose von Ausnahmefällen	81
4.3.2.1	Beschreibung von Ausnahmefällen	81
4.3.2.2	Detektion von Ausnahmefällen	83
4.3.3	Problem der Prognosegenauigkeit	87
4.4	Zusammenfassung des Kapitels	90
5	Handlungsalternativen im Ausnahmefall	91
5.1	Generelle Vorgehensweise zur Generierung von Handlungsalternativen	92
5.1.1	Ablauf der Handlungsalternativengenerierung	93
5.1.2	Steuerparameter für reale Materialflusssysteme	94
5.2	Einordnung und Abgrenzung der Problemstellung	96
5.2.1	Optimierung und Simulation	97
5.2.2	Fuzzy-Logik und Künstliche Neuronale Netze	98
5.2.3	Regelbasierte Systeme und Expertensysteme	99
5.2.4	Einordnung der Handlungsalternativengenerierung	100

5.3	Ein Verfahren zur Ermittlung und Bewertung von Handlungsalternativen	101
5.3.1	Mehrstufige Vorgehensweise der Handlungsalternativengenerierung . . .	102
5.3.2	Darstellung von Handlungsalternativen	104
5.3.3	Qualitative und zeitliche Belegungsplanung	105
5.3.4	Quantitative Steuerparameter und Sonderfälle	107
5.3.5	Ein allgemeiner Algorithmus zur Handlungsalternativengenerierung . . .	108
5.3.6	Adaption der Korrelationskoeffizienten an ein Materialflusssystem . . .	112
5.3.7	Einführung von Abbruchkriterien	114
5.4	Zusammenfassung des Kapitels	115
6	Prototypische Anwendung Simulationsbasierter Frühwarnsysteme	117
6.1	Bewertung der technischen Umsetzung	117
6.1.1	Verteilte Systeme, Web Services und XML-Datenbanken	117
6.1.2	Datenbeschaffung und -versorgung im praktischen Umfeld	119
6.1.3	Verwendete Simulatoren	120
6.2	Beschreibung der prototypischen Anwendungen	121
6.2.1	Die Frontschaltgetriebemontage	121
6.2.1.1	Beschreibung des Materialflusssystems	121
6.2.1.2	Simulationsmodell und Datenversorgung	122
6.2.1.3	Ausnahmedetektion und Handlungsalternativengenerierung . . .	123
6.2.1.4	Ergebnisse des SEWS für die FSG-Montage	125
6.2.2	Der Härteofen A28	127
6.2.2.1	Beschreibung des Materialflusssystems	127
6.2.2.2	Simulationsmodell und Datenversorgung	128
6.2.2.3	Ausnahmedetektion und Handlungsalternativengenerierung . . .	129
6.2.2.4	Ergebnisse des SEWS für den Härteofen A28	131
6.3	Zusammenfassung des Kapitels	134
7	Zusammenfassung und Ausblick	135
7.1	Zusammenfassung der Arbeit	135
7.2	Kritische Würdigung	137
7.3	Schlussfolgerungen und Ausblick	138
A	Die einzelnen Elemente des SEWS-spezifischen XML-Schemas	140
B	Performancevergleich zwischen AutoMod und SLX	145
C	Beispielhafte Darstellung einer Berechnungsregel	147
	Abkürzungsverzeichnis	150
	Literaturverzeichnis	151

Abbildungsverzeichnis

2.1	Zusammenhang System - Modell ([Obe99])	7
2.2	Simulation als interdisziplinäres Aufgabengebiet	8
2.3	Klassifikation der Simulation, in Anlehnung an ([Bac97])	10
2.4	Durchgängige Datenbasis mit Hilfe der digitalen Fabrik ([LGW99])	18
2.5	Anwendungsfelder der Simulation ([VDI00])	20
2.6	Komponenten eines Simulators ([VDI00])	21
3.1	Einordnung des Themengebietes SEWS	38
3.2	Die Hauptkomponenten eines Simulationsbasierten Frühwarnsystems	40
3.3	Komponenten des Frameworks	44
3.4	Verteilte Organisation der Komponenten eines SEWS	48
3.5	Ablaufplan des Workflows von SEWS	51
4.1	Simulationsrelevante Daten im traditionellen Sinne ([RNJ03])	58
4.2	Klassifikation der simulations- und SEWS-relevanten Daten	61
4.3	Zusammenhang der Kriterien zur Auswahl von IT-Systemen	63
4.4	XML-Schema für Simulationsbasierte Frühwarnsysteme	65
4.5	XML-Schema zur Verwaltung von XPath- und XUUpdate-Befehlen	67
4.6	Komponenten der allgemeinen XML-Schnittstelle ([JH06])	69
4.7	Ablauf und Funktionen zur Integration der Simulation in SEWS	76
4.8	XML Schema Definition zur Kennwertberechnung	82
4.9	Bestandsverlauf eines Puffers mit Grenzwertüberschreitung	85
4.10	Prinzipskizze der statistischen Verteilung der ZAZ relevanter Z^*	87
5.1	Aufbau der Komponente Generator of Alternatives	94
5.2	Klassierung möglicher Steuerparameter eines Materialflusssystems	95
5.3	Generelle Architektur eines traditionellen SO-Systems ([PGN06])	98
5.4	Differenzen und Gemeinsamkeiten mit anderen Lösungsverfahren	102
5.5	Das Kind-Element <i>ProcessSequences</i> des XML-Schemas für SEWS	104
5.6	Das Kind-Element <i>WorkOrderList</i> des XML-Schemas für SEWS	105
5.7	Allgemeiner Algorithmus zur Handlungsalternativengenerierung in SEWS	111
6.1	Benutzerschnittstelle zur Verwaltung Simulationsbasierter Frühwarnsysteme	118
6.2	Schematisches Layout der FSG-Montage	121
6.3	Transformiertes ASCII-File mit Ressourceninformationen	123
6.4	Auswertung der Mittleren Absoluten Abweichung für die FSG-Montage	126
6.5	Auswertung der durchschnittlichen Werkerauslastung in der FSG-Montage	127
6.6	Schematisches Layout des Härteofens A28	128
6.7	Auswertung der Mittleren Absoluten Abweichung des Härteofens A28	132

6.8	Auswertung des Bestandes an Rosten im Materialflusssystem Härteofen A28	133
A.1	Das Kind-Element <ProcessSystem> des XML-Schemas für SEWS	141
A.2	Das Kind-Element <Resources> des XML-Schemas für SEWS	142
A.3	Das Kind-Element <MeasuringPoints> des XML-Schemas für SEWS	143
A.4	Das Kind-Element <Products> des XML-Schemas für SEWS	143
A.5	Das Kind-Element <ShiftModels> des XML-Schemas für SEWS	144
C.1	Gerichteter Baum der Variable x	147

Tabellenverzeichnis

5.1	Mapping von Ausnahmefällen auf die Restriktionen der Belegungsplanung . .	107
5.2	Korrelationskoeffizienten für Ausnahmefälle und Steuerparameter	108
6.1	Korrelationskoeffizienten für die FSG-Montage	124
6.2	Mappingmatrix für den Härteofen A28	130
B.1	Performancevergleich zwischen AutoMod und SLX	145

Kapitel 1

Einführung

Aus kleinem Anfang entspringen alle Dinge.
— Cicero

1.1 Motivation

Die Automobilindustrie ist von einer Globalisierung der Märkte geprägt. Ein damit einhergehender Wandel der Technologien und die erhöhte Mobilität der Produktionsfaktoren bewirkt eine zunehmende Wettbewerbsintensität, Transparenz der Märkte und weltweite Überkapazitäten. Dadurch hat sich der Verkäufermarkt weitestgehend in einen Käufermarkt gewandelt. Die Automobilkonzerne reagieren auf diese Herausforderungen mit kürzeren Produktlebenszyklen, verstärkter Kundenorientierung, einer Individualisierung von Produktion und Leistung und somit einem größeren Produktportfolio. Es findet ein zeitlicher Wettlauf um Produkt- und Prozessinnovationen statt, wobei sich diese Entwicklungen in Form höherer Anforderungen in der Produktentstehungs- und Produktionsplanung bemerkbar machen.

Auch die operative Produktionssteuerung und -planung ist dadurch hohen Anforderungen ausgesetzt. Eine große Variantenvielfalt und die verstärkte Kundenorientierung bewirken eine steigende Komplexität und Dynamik. Dem dadurch steigenden Kostendruck soll mit möglichst geringen Beständen bei kurzen Durchlaufzeiten und einer höheren Flexibilität der Materialflusssysteme entgegen gewirkt werden. Dies bewirkt eine zunehmende Unsicherheit bei der Produktionsprogrammplanung. Die Forderung nach geringen Beständen setzt eine kurzfristige und schnelle Reaktionsmöglichkeit der Systeme voraus, da sich Störgrößen wie beispielsweise Maschinenausfälle oder kurzfristige Veränderungen des Produktionsprogramms sehr viel stärker auswirken und die Prozesssicherheit des Materialflusssystems gefährden.

Die Vorhersage von Ereignissen, Zuständen oder Entwicklungen ermöglicht einen zeitlichen Vorteil, da frühzeitig geeignete Reaktionen oder Gegenmaßnahmen erfolgen können. In einem Materialflusssystem sind ständig kurzfristige Entscheidungen zu treffen. In diesem kurzfristigen Umfeld ist hierzu ein Paradigmenwechsel von einer rein reaktiven zu einer proaktiven Steuerung zu vollziehen. Dabei beinhaltet die Reaktivität Handlungen, die auf vorangegangene Aktionen oder den aktuellen Systemzustand angepasst sind. Dagegen berücksichtigt proaktives Verhalten ein vorausschauendes und zielgerichtetes Handeln als Reaktion auf mögliche oder reale Zustandsänderungen. Aufgrund der Komplexität realer Materialflusssysteme ist die

Simulation besonders zur Prognose künftiger Systemzustände geeignet. Die Einführung Simulationsbasierter Frühwarnsysteme unterstützt die proaktive Steuerung von Materialflusssystemen. Dabei wird ein zeitlicher Vorteil sowohl durch die frühzeitige Erkennung von Ausnahmesituationen als auch durch die Selektion und Bewertung möglicher proaktiver Steuerungsmaßnahmen oder Handlungsalternativen erreicht.

Fowler und Rose diskutieren die künftigen Herausforderungen in der Modellierung und Simulation komplexer Produktionssysteme ([FR04]). Hierzu gehören unter anderem die Reduzierung des zeitlichen Aufwandes bei der Durchführung von Simulationsstudien. Die Zeit zur Informations- und Datenbeschaffung kann durch eine effiziente und automatisierte Datenaufbereitung, der Verbesserung der Datenqualität im industriellen Umfeld und durch die Verwendung standardisierter Datenaustauschformate geschehen. Darüber hinaus sind auch Reduzierungen des Aufwandes beim Modelldesign, bei der Modellentwicklung und dem Modelleinsatz möglich.

Eine weitere Tendenz ist die vermehrte Verwendung der Simulation als eingebettete Applikation in Produktionssteuerungs-, Decision-Support- oder Scheduling-Systemen ([Bru00]). Darüber hinaus gibt es Bestrebungen, die Simulation als Dienstleistung über das Web zur Verfügung zu stellen. In diesem Zusammenhang ist auch die Web-basierte Simulation zu nennen, worunter im engeren Sinne die Ausführung von Simulationsmodellen auf einem Webserver oder deren Ausführung als Applet in einem Web-Browser verstanden werden kann.

Weitere Erleichterungen für den Simulationsanwender sehen Fowler und Rose in der Entwicklung von Methoden zur schnellen Bereitstellung von Simulationsressourcen ([FR04]). Dies sind beispielsweise Simulationsmodelle der realen Produktion, die permanent zu aktualisieren und zu synchronisieren sind. Wird ein weiteres Simulationsmodell zur Entscheidungsunterstützung benötigt, so muss lediglich ein Klon von diesem Ursprungsmodell erzeugt werden. Eine andere Möglichkeit besteht in der automatischen Modellgenerierung, die bei Bedarf Simulationsmodelle aus den Fabrikdatenbanken erzeugt, die anschließend von einem Simulationsanwender zu Experimenten herangezogen werden können. Die schnelle und einfache Verknüpfung von Simulationsmodellen und anderen Support-Systemen wie bspw. Produktionsleitständen, Analysewerkzeugen oder Schedulingssystemen ist eine weitere künftige Herausforderung.

Auch bei Simulationsbasierten Frühwarnsystemen wird die Simulation als eingebettete Anwendung zur Entscheidungsunterstützung verwendet. Hierzu sind Simulationsmodelle erforderlich, die ein reales Produktionssystem abbilden und eine Frühwarnung vor unerwünschten Ereignissen ermöglichen. Diese Modelle müssen mit den realen Systemzuständen versorgt werden. Weiterhin sind mehrere Funktionskomponenten und Anwendungen in einer offenen und erweiterbaren Architektur miteinander zu verknüpfen, die sich harmonisch in die bestehende IT-Infrastruktur eines Unternehmens einfügen lassen.

1.2 Fragestellung und Zielsetzung

Die klassische Anwendung von Simulationsmodellen in Verbindung mit Produktions- und Logistiksystemen ist die Unterstützung bei der Planung, der Realisierung und dem Betrieb von Materialflusssystemen mit dem Ziel, deren komplexe und dynamische Zusammenhänge zu bewerten. Basierend auf dem gegenwärtigen Systemzustand des realen Materialflusssystemes müssen mit Hilfe der Simulation künftige kurzfristige Systemzustände vorhergesagt werden. Dabei ergeben sich noch ungelöste Fragestellungen, die in dieser Arbeit zu diskutieren sind.

Es bedarf einer Einordnung und Abgrenzung Simulationsbasierter Frühwarnsysteme und die Darstellung der notwendigen Funktionalitäten, sowie Komponenten, um zum einen die Simulation als eingebettete Applikation zu verwenden und zum anderen die effiziente Integration in eine bestehende Systemwelt zu ermöglichen. Es existiert keine allgemeingültige Architektur, die unabhängig vom realen System, den verwendeten Simulatoren, der Modellierungsphilosophie und der technischen Implementierung diese Funktionalitäten und Komponenten umfassend beschreibt. Hierbei ist zu klären, wie die Komplexität der Simulation und Integration der einzelnen Funktionen und Komponenten vor dem Anwender verborgen werden kann.

Simulationsmodelle müssen mit dem gegenwärtigen Systemzustand des realen Systems initialisiert werden. Dabei ergibt sich die Fragestellung, wie diese Synchronisation erfolgen kann und wie eine Einbindung der Informationssysteme des realen Systems zu gestalten ist, um eine flexible und effiziente Datenerfassung und die Anbindung Simulationsbasierter Frühwarnsysteme an Fabrikdatenbanken und Produktivsysteme zu ermöglichen. Dabei ist der Schwerpunkt auf die Standardisierung des Datenaustausches zu legen.

Die Anwendung der Simulation zur Prognose künftiger Systemzustände, um die operative Produktionssteuerung und -planung zu unterstützen, stellt besondere Anforderungen an die verwendeten Simulatoren und die Entwicklung von Simulationsmodellen. Diese sind herauszuarbeiten, wobei verstärkt auf die Unabhängigkeit von spezifischen Simulatoren und Simulationsmodellen zu achten ist.

Bei der Prognose künftiger Systemzustände steht die Detektion möglicher Ausnahmefälle wie beispielsweise zu hohe Bestände, Terminabweichungen oder Kapazitätsengpässe im Vordergrund. Dabei ist zu klären, was unter Ausnahmefällen respektive einer Frühwarnung zu verstehen ist und wie diese beschrieben werden können. Ähnliche Fragestellungen ergeben sich auch bei strategischen Frühwarnsystemen im Bereich logistischer Netzwerke oder der Absicherung der operativen Materialflussplanung in Supply-Chains. In diesem Zusammenhang ist die Prognosegenauigkeit von großer Bedeutung für die Akzeptanz eines Simulationsbasierten Frühwarnsystems, da diese mit zunehmender Anzahl von Fehlalarmen abnimmt.

Werden Ausnahmesituationen in einem Materialflusssystem festgestellt, so wird der zeitliche Vorteil dieser Früherkennung durch eine effiziente Lokalisierung geeigneter proaktiver Steuerungsmaßnahmen verstärkt. In der Praxis bereits erfolgreich eingesetzte Monitoringsysteme verwenden vordefinierte Handlungsalternativen für vordefinierte Ausnahmesituationen. Es stellt sich die Frage, ob ein umfassender Ansatz existiert, der auch neue, noch undefinierte Alternativen liefern kann, die auf die Art und den Umfang der Ausnahmesituation abgestimmt sind.

Eine letzte nicht zu vernachlässigende Zielsetzung liegt in einer möglichst effizienten, einfachen und offenen Implementierung unabhängig von einer unternehmensspezifischen IT-Infrastruktur. Weiterhin sind reale Anwendungsfälle von Interesse, die sowohl die An- und Umsetzbarkeit als auch die Potenziale Simulationsbasierter Frühwarnsysteme aufzeigen.

1.3 Aufbau der Arbeit

In einem ersten Kapitel wird auf die Computersimulation eingegangen. Dabei sind zunächst die geschichtliche Entwicklung, grundsätzliche Rahmenbedingungen für deren Einsatz, sowie Definitionen wichtiger Begrifflichkeiten interessante Themenschwerpunkte. Weiterhin erfolgt eine Klassifizierung der Simulation und eine Vorstellung grundsätzlicher Methoden, sowie die Darstellung der Anwendungsbereiche, des Nutzens und der Grenzen der Simulation. Anschließend wird die Simulation im Kontext der Automobilindustrie betrachtet. Die Haupteinsatzgebiete der Simulation liegen sowohl in der Planung von Produktionsanlagen, Logistikbereichen und -konzepten als auch in der operativen Produktionsplanung und -steuerung. In diesem Zuge sind besonders die informationstechnische Integration der Simulation in einem industriellen Umfeld, der Begriff der Digitalen Fabrik und der Einsatz der Simulation speziell in den Bereichen Produktion und Logistik von Interesse. Auf das Themengebiet der Simulationswerkzeuge wird ebenso eingegangen wie auf die Vorgehensweise bei der Modellerstellung. Zum Abschluss dieses ersten Kapitels werden Entwicklungstendenzen der Simulation in Produktion und Logistik diskutiert und im Zusammenhang mit Simulationsbasierten Frühwarnsystemen betrachtet.

Ein weiteres Kapitel beschäftigt sich mit der Thematik der Simulationsbasierten Frühwarnsysteme. Hierzu werden allgemeine Frühwarnsysteme anhand ihrer grundlegenden Funktionalitäten erläutert und auf strategische Frühwarnsysteme in Unternehmen eingegangen. Anschließend befasst sich das Kapitel speziell mit Simulationsbasierten Frühwarnsystemen in Produktion und Logistik. Es existiert noch keine geeignete und umfassende Definition Simulationsbasierter Frühwarnsysteme in der Literatur. Aus diesem Grund wird eine Definition hergeleitet und zu stellende Anforderungen entwickelt, wonach eine Abgrenzung und Einordnung im Vergleich zu anderen simulationsbasierten Ansätzen erfolgt. Anschließend beschäftigt sich das Kapitel mit einer allgemeinen Architektur Simulationsbasierter Frühwarnsysteme, wozu deren einzelne Komponenten vorgestellt werden. Diese Komponenten übernehmen dabei vielfältige Funktionalitäten, die sich zum einen aus den Anforderungen an Simulationsbasierte Frühwarnsysteme und zum anderen aus der notwendigen Integration der Hauptkomponenten zu einer Gesamtarchitektur ergeben. Es wird auf verteilte Systeme und Kommunikationsprinzipien eingegangen, die bei der Entwicklung einer verteilten Architektur anwendbar sind. Web Services respektive eine serviceorientierte Architektur sind hierbei geeignete Technologien. Abschließend wird auf den Workflow Simulationsbasierter Frühwarnsysteme und deren Steuerungsmöglichkeiten eingegangen.

Das vierte Kapitel diskutiert die Prognose künftiger Systemzustände. Hierzu wird zunächst die Datenbeschaffung und -verwaltung betrachtet, wobei eine Unterscheidung von simulations- und SEWS-relevanten Daten erfolgt. Anschließend beschäftigt sich das Kapitel mit Kriterien

zur Systemauswahl, die im Zusammenhang mit der Datenbeschaffung bei der Auswahl geeigneter Informationssysteme und Datenquellen Anwendung finden. Auf die SEWS-interne Datenverwaltung mit Hilfe von XML und einer XML-Datenbank wird ebenso eingegangen, wie auf die eigentliche Datengewinnung aus originären Datenquellen. Die Verwendung der Simulation als eingebettete Applikation führt zu speziellen Anforderungen an SEWS. Bei der Prognose künftiger Zustände des realen Systems steht die Problematik der Initialisierung von Simulationsmodellen und der technischen Umsetzung der Detektion von Ausnahmefällen im Vordergrund. Weiterhin werden sowohl Möglichkeiten zur Modellaktualisierung als auch das Problem der Prognosegenauigkeit diskutiert.

Das nachfolgende Kapitel befasst sich mit der Generierung von Handlungsalternativen im Ausnahmefall, wozu eine generelle Vorgehensweise dargestellt wird. Die Schwerpunkte hierbei sind der Ablauf der Handlungsalternativengenerierung, die generellen Steuerparameter eines realen Materialflusssystemes und die Begründung der Notwendigkeit eines Lösungsverfahrens. Weiterhin wird die Einordnung und Abgrenzung der Problemstellung der Handlungsalternativengenerierung beim Auftreten eines Ausnahmefalls diskutiert. Hierzu greift das Kapitel die Technologien Fuzzy-Logik, Künstliche Neuronale Netze, Regelbasierte Systeme und Expertensysteme, sowie die Thematik der Simulation und Optimierung auf. Anschließend erfolgt die Darstellung eines Verfahrens zur Ermittlung und Bewertung von Handlungsalternativen für Simulationsbasierte Frühwarnsysteme. Dabei wird eine mehrstufige Vorgehensweise begründet, die Darstellung von Handlungsalternativen in der SEWS-internen Datenverwaltung erläutert und sowohl auf die zeitliche und qualitative Belegungsplanung als auch auf quantitative Steuerparameter und Sonderfälle eingegangen. Weiterhin erfolgt die Vorstellung eines allgemeinen Algorithmus zur Handlungsalternativengenerierung und die Adaption der Korrelationskoeffizienten an ein Materialflusssystem. Zum Abschluss werden Abbruchkriterien eingeführt, die eine Beendigung des Algorithmus garantieren.

Ein weiteres Kapitel diskutiert die prototypische Anwendbarkeit Simulationsbasierter Frühwarnsysteme. Hierzu erfolgt eine Bewertung der technischen Umsetzung im betrieblichen Umfeld eines Industrieunternehmens. Dabei wird sowohl der Einsatz verteilter Systeme, Web Services und XML-Datenbanken als auch die praktischen Problemstellungen bei der Datenbeschaffung und -versorgung und die verwendeten Simulatoren betrachtet. Es erfolgt die Erläuterung von Fallbeispielen, die in Form der FSG-Montage und dem Härteofen A28 eine Bewertung des Einsatzes Simulationsbasierter Frühwarnsysteme ermöglichen. Diese Beispiele werden explizit beschrieben, wobei jeweils die Datenversorgung, die Simulationsmodelle, die Ausnahmedetektion und Handlungsalternativengenerierung erörtert, sowie die konkreten Ergebnisse diskutiert werden. Es erfolgt eine abschließende Bewertung von SEWS mit ihren Nachteilen, Vorteilen und Potenzialen.

Eine Zusammenfassung, die kritische Würdigung der Arbeit und ein Ausblick mit Schlussfolgerungen runden die Arbeit in einem letzten Kapitel ab.

Kapitel 2

Simulation in Produktion und Logistik

Nur ein Narr macht keine Experimente.

— Charles Darwin

Dieses Kapitel befasst sich mit der Simulation in Produktion und Logistik. Hierzu wird zunächst auf die Computersimulation eingegangen, indem Definitionen, Klassifikationen und Methoden der Simulation ebenso diskutiert werden, wie deren Anwendungsgebiete, Nutzen und Grenzen. Abschließend ist die Simulation im Kontext der Automobilindustrie zu betrachten. Dabei sind vor allen Dingen die informationstechnische Integration der Simulation, die Digitale Fabrik und im Speziellen der Einsatz der Simulation in Produktion und Logistik bedeutsame Themengebiete.

Das Wort Simulation leitet sich aus dem lateinischen *similis* bzw. *simulatio* ab und bedeutet so viel wie „ähnlich“ bzw. „Täuschung“, „Verstellung“ ([Dot03]). Die Technik der Simulation ist sehr alt, da sich bereits frühe Kulturen der Simulation in Form vorzeitlicher Planetarien oder der Himmelscheibe von Nebra bedienten, um den Sonnengang nachzuvollziehen. Auch diese aus heutiger Sicht primitiven Konstruktionen stellen eine Form der Simulation dar.

In der Wissenschaft versteht man unter der Simulation im Allgemeinen die Nachbildung eines realen Objektes, Vorganges oder komplexen Systems in Form eines Modells. Dieses Modell wird zur Durchführung von Experimenten genutzt, um Erkenntnisse über das reale Objekt, den Vorgang oder das komplexe System zu gewinnen ([Hor03]). In der allgemeinen Modelltheorie ist nach Stachowiak ein Modell durch folgende drei Merkmale gekennzeichnet ([Sta73]):

- **Abbildung**
Ein Modell ist ein Abbild von etwas, eine Repräsentation natürlicher oder künstlicher Originale, die selbst wieder Modelle sein können.
- **Verkürzung**
Ein Modell erfasst nur die für den Modellnutzer bzw. Modellersteller relevanten Attribute des Originals.
- **Pragmatismus**
Die Modellerstellung erfolgt innerhalb einer bestimmten Zeitspanne zu einem bestimmten Zweck. Dabei werden bestimmte Merkmale vernachlässigt, um die wesentlichen Modelleigenschaften hervorzuheben. Das Modell wird also vom Original abstrahiert und es erfolgt eine Orientierung am Nützlichen.

Bei einem Vorgang handelt es sich allgemein um einen Prozess, der einen definierten Ablauf von Zuständen eines Systems darstellt. Unter einem System versteht man nach ([VDI00]) eine abgegrenzte Anordnung von Komponenten, die miteinander in Beziehung stehen.

Nach Oberweis simuliert ein Modell M ein System S , wenn durch die Umformung des Inputs I des Systems in den Input I' des Simulationsmodells und durch Umformung des zu I' produzierten Outputs O' der Output O des zu simulierenden Systems erhalten werden kann (siehe Abb. 2.1) ([Obe99]).

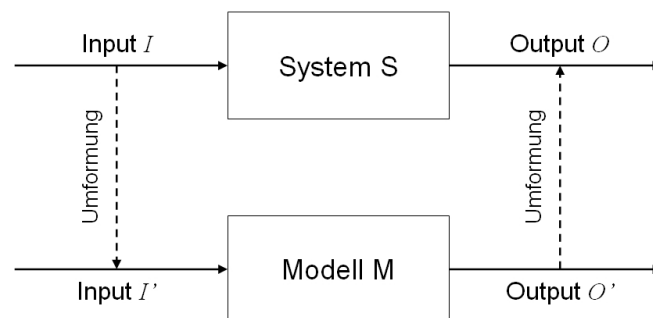


Abb. 2.1: Zusammenhang System - Modell ([Obe99])

Bei einem Simulationsexperiment werden wiederholt Simulationsläufe durchgeführt, wobei das Verhalten des Modells mit Hilfe einer systematischen Parameter- oder Strukturvariation gezielt empirisch untersucht werden soll ([VDI00]).

Im Zentrum dieser Arbeit steht der Einsatz der Simulation in der Produktion und Logistik. Dies ist jedoch ohne den Einsatz leistungsfähiger Rechnersysteme nicht denkbar. Aus diesem Grund soll im Folgenden näher auf die Computersimulation eingegangen werden, bei der sich die Simulation eines auf einem Computer ausführbaren Modells bedient.

2.1 Computersimulation

Aus Sicht der Informatik beginnt die Geschichte der Simulation in den 1940er Jahren, da hier die ersten mit veränderbarem Speicherinhalt ausgestatteten und frei programmierbaren Rechenmaschinen entwickelt wurden. Diese Rechenmaschinen dienten überwiegend zu Simulationszwecken, um das Verhalten des Wetters, von Flugkörpern, von radioaktiven Stoffen, von elektrischen Schaltkreisen, von natürlichen Himmelskörpern und künstlichen Satelliten zu simulieren ([Hor03]). Für die Quantenphysik wurde die Simulation unersetzbar. Insbesondere das Militär, die Ökonomie und Politik stützten sich zunehmend auf Simulationsmodelle, da die Komplexitätszunahme globaler ökonomischer, militärischer und politischer Prozesse der Prognose mit Hilfe der Simulationstechnik eine besondere Rolle zukommen ließ. So wurde beispielsweise mit Hilfe der Simulation die Entwicklung von Kernwaffen während des Manhattan Projektes unterstützt ([Wik07g]).

Heute ist die Simulation ein Standardwerkzeug für Ingenieure und Wissenschaftler. Sie ermöglicht die Durchführung von Experimenten in einem Rechner bzw. einem virtuellen Labor, um eine technisch-wissenschaftliche Aufgabenstellung durch Abstraktion von der Realität modellhaft vor der tatsächlichen Realisierung mit verschiedenen Parametern risikofrei und kostengünstig darzustellen, sowie Experimente durchzuführen ([KR98],[Gor03]). Als interdisziplinäres Aufgabengebiet befindet sie sich im Überlappungsbereich zwischen Angewandter Mathematik, Informatik und verschiedensten Anwendungsbereichen (siehe Abb. 2.2).

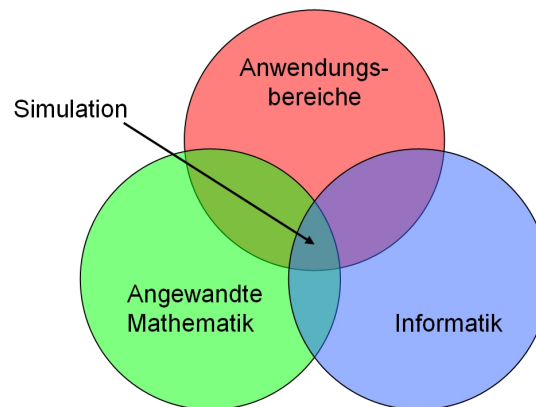


Abb. 2.2: Simulation als interdisziplinäres Aufgabengebiet

Der sinnvolle Einsatz der Simulation ist von den Rahmenbedingungen der Aufgabenstellung abhängig und besonders dann interessant, wenn einer der folgenden Punkte gegeben ist ([DD05]):

- Ein vollständiges mathematisches Modell ist nicht verfügbar und auch nicht mit vertretbaren Kosten entwickelbar.
- Die Anwendung verfügbarer, analytischer Methoden setzen vereinfachende Maßnahmen voraus, die den Kern eines vorliegenden Problems verfälschen können.
- Ein praktikabler Einsatz analytischer Methoden ist nicht möglich, beziehungsweise mit erheblichem Aufwand verbunden.
- Es ist zu komplex, kostspielig oder riskant, reale Experimente durchzuführen.

Der folgende Abschnitt erläutert Begrifflichkeiten zur Thematik der Simulation. Anschließend erfolgt eine Klassifikation und es werden verschiedene Methoden bei deren Einsatz vorgestellt. Zum Abschluss werden sowohl die Anwendungsbereiche der Simulation diskutiert als auch deren Nutzen und Grenzen aufgezeigt.

2.1.1 Definitionen zur Simulation

Dieser Abschnitt entwickelt ausgehend von der Simulation die Begrifflichkeiten Simulationsmodell, Simulator und Simulationsexperiment Schritt für Schritt. Dabei wird auf die wichtigsten Definitionen eingegangen.

Nach Brose ist die Simulation im Allgemeinen und übergeordneten Sinn

„[...] die modellhafte Darstellung oder Nachbildung bestimmter Aspekte eines vorhandenen oder zu entwickelnden [...] Systems [...], insbesondere auch seines Zeitverhaltens.“ ([Bro02])

Dabei wird unter dem Zeitverhalten die Entwicklung der dynamischen Prozesse eines Systems über die Zeit verstanden. Die VDI-Richtlinie 3633 berücksichtigt darüber hinaus die Experimentierfähigkeit eines Modells in ihrer Definition der Simulation, die im Rahmen dieser Arbeit Gültigkeit besitzen soll ([VDI00]):

„Simulation ist das Nachbilden eines Systems [...] in einem experimentierbaren Modell, um zu Erkenntnissen zu gelangen, die auf die Wirklichkeit übertragbar sind. [...] Im weiteren Sinne wird unter Simulation das Vorbereiten, Durchführen und Auswerten gezielter Experimente mit einem Simulationsmodell verstanden.“

Die Modellbildung ist offenbar eine wichtige und kreative Aufgabe, die der Darstellung der nicht exakt abzubildenden Realität mit Hilfe wissenschaftlicher Methoden durch Abstraktion und Vereinfachung entspricht ([AF05]). Grundsätzlich bestehen Simulationsmodelle aus statischen und dynamischen Objekten, die sowohl einzelne Systemkomponenten als auch ganze Klassen solcher Komponenten sein können. Diese Objekte beeinflussen sich gegenseitig und stehen in Wechselwirkung miteinander. Nach der VDI-Richtlinie 3633 ist ein Simulationsmodell

„[...] eine vereinfachende Nachbildung eines geplanten oder real existierenden Originalsystems und -prozesses in einem anderen begrifflichen oder gegenständlichen System. Das Modell unterscheidet sich hinsichtlich der untersuchungsrelevanten Eigenschaften nur innerhalb eines vom Untersuchungsziel abhängigen Toleranzrahmens vom Vorbild.“ ([VDI00])

Da sich diese Arbeit schwerpunktmäßig mit der Simulation in Produktion und Logistik beschäftigt, stellt ein Simulationsmodell ein formales, durch eine entsprechende Simulationssoftware ausführbares Modell dar. Dabei handelt es sich um einen Simulator, der nach der VDI-Richtlinie 3633 wie folgt definiert ist ([VDI00]):

„Softwareprogramm, mit dem ein Modell zur Nachbildung des dynamischen Verhaltens eines Systems und seiner Prozesse erstellt und ausführbar gemacht werden kann.“

Fasst man die Definitionen noch einmal zusammen, so ist ein Simulator die Implementierung oder Realisierung eines Simulationsmodells, welches wiederum eine Abstraktion des zu simulierenden Systems darstellt. Die Durchführung eines Simulationsexperimentes ist dabei der Ablauf des Simulationsmodells mit parametrisierten Eingangswerten. Die Ergebnisse werden anschließend interpretiert und die daraus gewonnenen Erkenntnisse werden auf das reale System übertragen.

Nachdem in diesem Abschnitt eine umfangreiche Definitionensammlung vorgestellt wurde, soll im Folgenden eine Klassifikation der Simulation erfolgen.

2.1.2 Klassifikation und Methoden der Simulation

In der Literatur finden sich verschiedene Ansätze zur Klassifikation der Simulation, und je nach Problemstellung kommen verschiedene Simulationsmethoden zur Anwendung. Eine grundlegende Einteilung der Simulation ist hinsichtlich folgender Merkmale möglich:

- Verwendete Eingangsgrößen
- Dynamik der Modelle
- Art der Zeitsteuerung

Ist mindestens eine der Eingangsgrößen zufallsabhängig, so handelt es sich um eine stochastische Simulation. Sind dagegen alle Eingangsgrößen zufallsunabhängig, oder können als solche betrachtet werden, so spricht man von einer deterministischen Simulation ([Bac97]).

Bei der Dynamik von Simulationsmodellen können statische und dynamische Modelle unterschieden werden, wobei der Unterschied in der Zeitabhängigkeit des Verhaltens liegt. Ein dynamisches Modell ändert seinen Zustand zeitabhängig, während statische Modelle ohne Zeitbezug betrachtet werden ([Pag91]). Grundsätzlich können sowohl dynamische als auch statische Modelle deterministisch oder stochastisch sein.

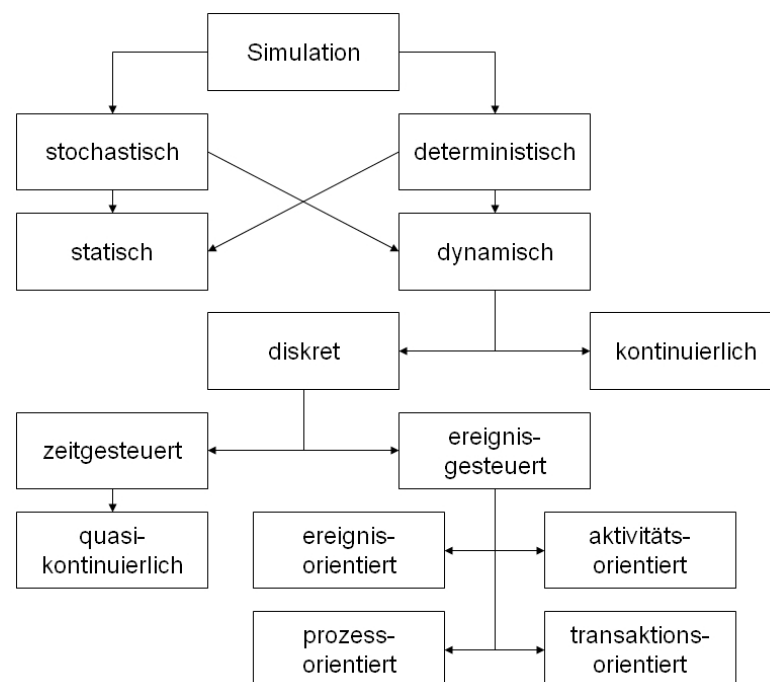


Abb. 2.3: Klassifikation der Simulation, in Anlehnung an ([Bac97])

Unterscheidet man Simulationsmodelle nach der Art der Zeitsteuerung, so gibt es Modelle ohne Zeitbezug, zeitkontinuierliche und zeitdiskrete Modelle. Bei den Modellen ohne Zeitbezug handelt es sich um die bereits angesprochenen statischen Modelle. Dagegen verändert sich der

Zustand der Modelle bei der zeitkontinuierlichen und zeitdiskreten Simulation mit dem Zeitfortschritt. Dabei wird der Zustand eines Modells durch die einzelnen Zustände seiner statischen und dynamischen Objekte beeinflusst und repräsentiert. Bei der zeitdiskreten Simulation können wiederum zeitgesteuerte und ereignisgesteuerte Modelle unterschieden werden.

Abbildung 2.3 stellt die vorgestellte Klassifikation in Anlehnung an Backes nochmals übersichtlich dar ([Bac97]). Im Bereich der Produktion und Logistik kommt vor allen Dingen die zeitdiskrete Simulation zum Einsatz, weshalb diese im Folgenden näher erläutert wird.

2.1.2.1 Diskrete Simulation

Wie bei der kontinuierlichen Simulation dient auch die diskrete Simulation der Modellierung und Analyse dynamischer Systeme ([DD05]). Dabei wird der Systemzustand durch zeitabhängige Zustandsvariablen beschrieben. Der Unterschied zur kontinuierlichen Simulation liegt in der Art und Weise, wie Zustandsänderungen erfolgen. Dies geschieht bei der diskreten Simulation nicht kontinuierlich, sondern sprunghaft zu endlich bestimmten Zeitpunkten, an denen Ereignisse eintreten. Die Festlegung dieser diskreten Zeitpunkte geschieht mit Hilfe eines so genannten *Schedulers*. Eine Simulationsuhr wird immer wieder auf den nächsten aktuellen Ereigniszeitpunkt gesetzt. Das Verhalten dazwischen wird nicht betrachtet und als konstant bzw. unverändert angenommen. Die zeitliche Entwicklung eines Systems wird durch eine endliche Folge von Zuständen abgebildet ([Fis01]).

Die Anwendungsbereiche der diskreten Simulation liegen vor allen Dingen in der Simulation von Fertigungs-, Transport- und Verkehrssystemen aufgrund der einfacheren Abbildung der Ablaufsteuerung im Vergleich zu anderen Modellierungsarten ([Pag91],[Ban05]). Zur Bestimmung der diskreten Ereigniszeitpunkte können zwei grundsätzliche Methoden zum Einsatz kommen, die im Folgenden näher erläutert werden. Es handelt sich dabei um die zeitgesteuerte diskrete und ereignisgesteuerte diskrete Simulation.

Zeitgesteuerte Diskrete Simulation

Bei der zeitgesteuerten diskreten Simulation wird die Modellzeit um äquidistante Zeitintervalle Δt erhöht. Das System wird in immer gleichen Zeitabständen betrachtet und somit in Zeitscheiben zerlegt. Aus diesem Grund wird die zeitgesteuerte diskrete Simulation auch als Zeitscheibensimulation bezeichnet.

Nach der Erhöhung der Modellzeit um das Inkrement Δt wird eine Aktualisierung des Systemzustandes durchgeführt, indem die in der letzten Epoche aufgetretenen Zustandsänderungen durchgeführt werden. Diese Aktualisierung wird auch als synchrone Aktualisierung bezeichnet. Wird das Δt infinitesimal klein gewählt, so kann ein quasi-kontinuierliches Simulationsmodell erhalten werden, welches die Ausführung kontinuierlicher Modelle auf Digitalrechnern ermöglicht.

Probleme bei der zeitgesteuerten diskreten Simulation liegen zum einen in der Bestimmung geeigneter Zeitintervalle und zum anderen in dem Auftreten so genannter *Totzeiten*. Der Zustand des Systems bleibt zu vielen Zeitpunkten unverändert und wird dennoch aufgrund des fixen zeitinkrements zu einem festgelegten Zeitpunkt aktualisiert.

Ereignisgesteuerte Diskrete Simulation

Unter einem Ereignis wird die Veränderung des Systemzustandes zu einem bestimmten Zeitpunkt, dem Ereigniszeitpunkt, verstanden ([Lie92],[Ban05]). Das Eintreten eines Ereignisses bewirkt eine Zustandsänderung innerhalb des Modells, wobei das Ereignis selbst keine Modellzeit in Anspruch nimmt. Zwischen zwei Ereignissen bleibt der Systemzustand unverändert bzw. konstant.

Nach dem Eintritt eines Ereignisses steht fest, welche Ereignisse künftig eintreten werden. Diese Ereignisse werden errechnet und in einer Ereignisliste gespeichert. Der Vorteil gegenüber der zeitgesteuerten diskreten Simulation liegt in der Einsparung von Rechenzeit, da direkt mit dem nächsten Ereignis fortgefahren werden kann, ohne die dazwischen liegende Totzeit zu berücksichtigen.

Der Großteil der kommerziellen Simulatoren für den Einsatz in Produktion und Logistik basieren auf der ereignisorientierten Zeitsteuerung. Man findet in der Literatur verschiedene Ansätze bezüglich der Simulationsmethoden im Bereich der ereignisgesteuerten diskreten Simulation. Eine grundlegende Klassifikation ist hinsichtlich der Darstellung von Ereignissen und Zuständen der realen Welt in einem Simulationsmodell möglich ([Hoo86],[Ban05]):

- Ereignisorientiert
- Aktivitätsorientiert
- Prozessorientiert
- Transaktionsorientiert

Im folgenden Abschnitt werden sowohl die Anwendungsbereiche der Simulation als auch deren Grenzen aufgezeigt. Dabei wird zunächst auf Vor- und Nachteile der Simulation eingegangen, bevor die Simulation in der Automobilindustrie diskutiert wird.

2.1.3 Anwendungsbereiche, Nutzen und Grenzen

Die Anwendungsbereiche der Simulation sind sehr vielfältig. Sie reichen vom Einsatz in ökonomischen Bereichen über die Simulation von chemischen und physikalischen Abläufen in der Verfahrensindustrie, der Optimierung von Materialflüssen und logistischen Problemstellungen in der Automobilindustrie bis hin zum Einsatz von Flug- und Fahrsimulatoren in der Luft- und Raumfahrttechnik. Auch zur Prognose und Analyse von Entwicklungsszenarien von Volkswirtschaften oder klimatischen Umweltbedingungen kommt die Simulation zum Einsatz. Zusammenfassend können folgende Einsatzfelder der Simulation genannt werden ([KG95]):

- Entscheidungsunterstützung
- Planung
- Forschung und Entwicklung
- Organisationsgestaltung
- Ausbildung und Training

Die Vorteile des Einsatzes der Simulation liegen durch ihre modulare Struktur klar auf der Hand. Simulationsmodelle können flexibel an spezifische Fragestellungen angepasst werden. Unabhängig von der Systemart ist es möglich, mit einheitlichen Methoden und vielseitig verwendbaren Softwarewerkzeugen zu arbeiten ([Bos04]).

Wird die Simulation bereits in der Planungs- und Projektierungsphase angewendet, so können Schwachstellen frühzeitig aufgezeigt und vermieden werden. Weiterhin ist eine Alternativenbewertung durch Veränderungen am Modell ohne negative Auswirkungen auf ein reales System oder gar dessen Gefährdung möglich ([Bac97],[Sch03]). Im Vergleich zu realen oder analogen physikalischen Modellen beträgt der finanzielle Aufwand bei der rechnergestützten Modellerstellung nur ein Bruchteil.

Viele Simulatoren ermöglichen eine Visualisierung bzw. Animation des realen Systems, was das Problemverständnis und die Akzeptanz fördert. Der simulierte Prozess kann durch Zeitrafung oder Zeitdehnung beschleunigt oder verlangsamt werden. Dadurch können Vorgänge, die einige Jahre benötigen würden, in wenigen Sekunden ablaufen. Umgekehrt sind schnelle Prozesse, die in Sekundenbruchteilen beendet sind, über Minuten verzögerbar. Aus diesen Gründen ist die Simulation besonders für Schulungs- und Ausbildungszwecke interessant, wobei die Realsysteme wie bspw. Flugzeuge bei der Ausbildung von Piloten nicht gefährdet werden.

Der Einsatz der Simulation hat aber auch seine Schwachstellen. Die Ergebnisse einer Simulationsstudie sind nur so gut wie die verwendeten Eingangsdaten bzw. Datengrundlage für das Simulationsmodell. Die Datenerfassung und -aufbereitung ist mit einem hohen Zeitaufwand verbunden, da die erforderlichen und oftmals erheblichen Datenmengen meist nicht oder nicht in geeigneter Form zur Verfügung stehen. Weiterhin ist bei der Modellierung eine Abstraktion vom realen System und das Treffen vereinfachender Annahmen notwendig.

Die Anforderungen an einen Simulationsexperten sind hoch. Er muss sowohl über technisches Verständnis zur Durchführung einer Analyse des realen Systems als auch über Erfahrungen in dem Bereich der Informatik verfügen. Darüber hinaus muss er Kenntnisse von wirtschaftswissenschaftlichen Zusammenhängen besitzen und in der Lage sein, statistische Auswertungen zu erstellen und zu analysieren. Die Aussagekraft eines Simulationsmodells ist sehr stark von der Erfahrung des Simulationsexperten abhängig, da die Ergebnisse von Simulationsexperimenten in einem gewissem Rahmen interpretiert werden müssen ([Sch01]). Die Simulationsmodelle zweier Experten sind aufgrund individueller Modellierungsphilosophien und -eigenarten nicht identisch, obwohl die Ergebnisse ähnlich oder gleich sein können. Mängel bei der Datenerhebung und der Interpretation von Ergebnissen können schnell zu falschen Entscheidungen führen.

Weiterhin besteht die Gefahr, dass bei größeren Problemstellungen der Arbeitsaufwand zur Modellentwicklung überproportional ansteigt. Der Grund liegt in der Zunahme der Permutationen und Freiheitsgrade von zu berücksichtigenden Parametern.

Bei zufallsbedingten Eingangsgrößen muss darüber hinaus eine Vielzahl identischer und unabhängiger Experimente durchgeführt werden, um zu einem statistisch gesicherten Ergebnis zu kommen. Die Ziele einer Simulationsstudie sind meist nicht harmonisch, sondern von konkurrierender Natur.

2.2 Simulation in der Automobilindustrie

Der Schwerpunkt dieser Arbeit in Bezug auf die Simulation liegt in der Anwendung der Simulation im Bereich von Produktion und Logistik. In der Automobilindustrie liegt das Haupteinsatzgebiet in der Planung von Produktionsanlagen, Logistikbereichen und -konzepten. Auch in der operativen Produktionsplanung und -steuerung kommt die Simulation vermehrt zum Einsatz ([Bay03]).

Durch Herausforderungen wie kürzere Produktlebenszyklen, verstärkte Kundenorientierung, einer Individualisierung von Produktion und Leistung und einem größeren Produktportfolio ist die Automobilindustrie einem scharfen Wettbewerb unterworfen. Diese Umstände machen es notwendig, nach effizienten Methodiken zu suchen, um sowohl die Planungsqualität zu verbessern, den Planungsablauf effizienter zu gestalten und die Planungsdauer bei Neuproduktprojekten zu beschleunigen. Auch die operative Produktionssteuerung und -planung sind aufgrund größerer Variantenvielfalt und verstärkter Kundenorientierung höheren Anforderungen durch steigende Komplexität und Dynamik in der Produktion und Logistik ausgesetzt ([Die01]).

Die Kundenbedürfnisse rücken immer stärker in den Mittelpunkt, wobei die Leistungserbringung zeitgerecht, qualitativ hochwertig und kostengünstig zu erbringen ist. Neben dem zeitlichen Wettlauf um Produkt- und Prozessinnovationen verschärft sich also auch der Wettbewerb um Zeit, Kosten und Qualität ([DD05],[VDI00]). Mit Hilfe neuer Methoden in der industriellen Produktionswirtschaft bspw. in Form der Digitalen Fabrik, versucht man dieser Entwicklung entgegen zu treten. Die Simulation als Teilbereich der Digitalen Fabrik kann hierbei mit den bereits vorgestellten Vorteilen unterstützen.

Die Integration in den Planungsalltag gestaltet sich jedoch problematisch. So muss sich die Simulation immer wieder hinsichtlich ihrem effektiven Nutzen, dem kostenmäßigen Mehraufwand und einer möglichen Einbettung in den standardisierten Planungsablauf rechtfertigen, obwohl Studien die Wirtschaftlichkeit eindeutig nachweisen. Gora beschreibt die Einsparungsmöglichkeiten durch den Einsatz der Simulation in der Planung ([Gor03]). Bei Neuplanungen ist es möglich, bis zu 20 % der geplanten Investitionssummen einzusparen. Produktivitätsverbesserungen von 15 - 20 % sind bei bereits vorhandenen Materialflusssystemen möglich. Der Bestand und die Durchlaufzeiten können um bis zu 20 - 60 % reduziert werden.

Die Simulation stellt hohe Anforderungen an die beteiligten Personen. Verschiedene Unternehmensbereiche müssen kollaborativ zusammenarbeiten, um die Simulation erfolgreich an-

zuwenden. Darüber hinaus muss der Simulationsexperte meist Unterstützungs- und Überzeugungsarbeit gegenüber dem Management leisten ([Jak91]).

In dem folgenden Abschnitt wird zunächst auf die informationstechnische Integration der Simulation in produzierenden Unternehmen eingegangen. Die wichtigsten Technologien und Methodiken zur Datenverwaltung und Unterstützung der Produktion in einem Unternehmen werden vorgestellt. Aufgrund seiner Sonderstellung in der Automobilindustrie wird anschließend der Begriff „Digitale Fabrik“ näher betrachtet, welcher die Simulation als Teilbereich eines übergreifenden Konzeptes umfasst. Abschließend beschäftigt sich dieser Abschnitt explizit mit der Simulation in Produktion und Logistik.

2.2.1 Informationstechnische Integration der Simulation

Die Simulation benötigt erhebliche Datenmengen, die in verschiedenen Informationssystemen verwaltet werden. Nach Domschke sind dies Informationen über Anlagen, Produkt- und Produktionsinformationen, das Fabrikmodell und ablauforganisatorische Informationen ([DD05]). Bereits seit den 1960er Jahren werden Rechnersysteme zur Unterstützung der Produktion in den Unternehmen eingesetzt und ständig weiter entwickelt. Trotz des technischen Fortschritts benötigt ein Simulationsexperte ca. 40 % des zeitlichen Aufwandes für eine Simulationsstudie zur Datenbeschaffung ([Wor02b]).

Bei der Modellierung ist das dynamische Prozessverhalten des abzubildenden Systems darzustellen. In Abhängigkeit dieses dynamischen Prozessverhaltens wird eine Vielzahl an Informationen benötigt, die einen oder mehrere der folgenden Datenumfänge beinhalten können ([AF05]):

- Daten über das Material (bspw. Waren, Teiletypen und Menge).
- Daten über vorhandene Fördermittel und Lageranlagen wie Kapazitäten, Takt- und Bedienzeiten.
- Produktionstechnische Daten wie Fertigungstechnik, Fertigungseinrichtungen, Schnittstellen und DV-Technik zum Materialflusssystem.
- Daten des externen Warentransports wie Lieferanten- und Kundenbeziehungen, physische und DV-Schnittstellen im Wareneingang und im Warenausgang.
- Betriebswirtschaftliche Daten wie Investitionen, Kapitalbindungs- und Betriebskosten.
- Gegebenenfalls sonstige Daten über Gebäude, Grundstücke oder der Unternehmensstrategien.

Im Folgenden sollen aktuelle Systeme, Methoden und Techniken zur Informationsverarbeitung in einem Unternehmen vorgestellt werden, auf die bei der Durchführung von Simulationsstudien nicht verzichtet werden kann.

2.2.1.1 Computer Integrated Manufacturing

Das *Computer Integrated Manufacturing (CIM)* ist der integrative EDV-Einsatz in allen mit der Produktion zusammenhängenden Bereichen, der schon seit 1965 bekannt und verbreitet ist. Es handelt sich dabei um eine komplexe Methodik, die viele verschiedene Einflussfaktoren und Aspekte berücksichtigt ([Wik07a]).

Die Grundidee des CIM ist die direkte Verbindung der informationsverarbeitenden und steuernden Rechner untereinander, um sowohl manuelle Fehler bei der Bearbeitung und Übertragung von Daten zu vermeiden als auch den Informationsaustausch zwischen diesen Systemen zu beschleunigen. Weiterhin waren schnellere Angebotsabgaben, genauere Kalkulationen, höhere Termintreue, eine größere Flexibilität in Fertigung und Planung, sowie eine transparentere Kostenrechnung wichtige Zielsetzungen ([Kie98]).

Der ursprüngliche Begriff des CIM bezieht sich auf die Belange der Produktion. Es zeigte sich aber schnell, dass CIM nicht nur technische Aufgaben umfasst, sondern sowohl betriebswirtschaftlich-dispositive Aufgaben der Produktionsplanung und -steuerung als auch zwischen- und überbetriebliche Informationsflüsse beinhaltet ([BR93]).

Der Ausschuss für wirtschaftliche Fertigung gab 1985 folgende Definition für das CIM an ([AWF85]):

„CIM beschreibt den integrierten EDV-Einsatz in allen mit der Produktion zusammenhängenden Betriebsbereichen. Es umfasst das informationstechnische Zusammenwirken zwischen CAD, CAP, CAM, CAQ und PPS. Hierbei soll die Integration der technischen und organisatorischen Funktionen zur Produkterstellung erreicht werden. Dies bedingt die gemeinsame Nutzung aller Daten eines EDV-Systems, auch Datenbasis genannt.“

Die für die Simulation relevanten Informationen sind überwiegend in den Bereichen PPS und BDE zu finden, worauf im Folgenden näher eingegangen wird.

2.2.1.2 Produktionsplanungs- und Steuerungssysteme

Die *Produktionsplanung und -steuerung (PPS)* ist ein wesentlicher Bestandteil des *Computer Integrated Manufacturing*. Sie sind das Bindeglied zwischen den produktionswirtschaftlichen und produktionstechnischen Funktionen eines Unternehmens der produzierenden Industrie ([JB98]).

Die von einem PPS-System zu erfüllenden Aufgaben werden in Kern- und Querschnittsaufgaben untergliedert. Zu den Querschnittsaufgaben gehören die Auftragskoordination, das Lagerwesen und das PPS-Controlling. Die Kernaufgaben umfassen die Produktionsprogrammplanung, die Materialbedarfs- und Personaleinsatzplanung, die Auftragsplanung, sowie die Produktionssteuerung zur Durchführung und Überwachung der Produktion. Dabei beinhaltet die Produktionssteuerung mit ihrem operativen Charakter die endgültige Kapazitätsbelegung, den

Personaleinsatz und die Materialeinsatzsteuerung. Eine weitere Teilaufgabe der Produktionssteuerung liegt in der Produktionsüberwachung, wozu die auftragsorientierte Mengen- und Zeitkontrolle, sowie die zeitliche Bedarfsplanung gehören ([Ste03]).

PPS-Systeme sind auf die mittelfristige Produktionsplanung und -steuerung ausgerichtet und berücksichtigen gegebene Eigenarten spezifischer Materialflusssysteme nicht. Aus diesem Grund weisen sie Mängel in der kurzfristigen Produktionsplanung und -steuerung auf. Dadurch konnten sich Fertigungsleitstände etablieren, die systemspezifische Eigenarten berücksichtigen und die operative Produktionsebene mit der mittelfristigen Planungsebene verbinden ([Mai92]).

2.2.1.3 Die Betriebsdatenerfassung

Die konsequente Umsetzung der gewünschten Rationalität von PPS-Systemen wird auf der operativen Ebene durch die so genannte *Betriebsdatenerfassung (BDE)* erreicht. Dabei übernimmt die BDE keine Steuerungsfunktionen, sondern dient lediglich der Erfassung und Analyse von Betriebsdaten zur laufenden Überwachung (*Monitoring*). Zu den Betriebsdaten einer BDE zählen ([FFG97]):

- Auftragsdaten (bspw. Fertigungsstückzahlen und Fertigungszeiten)
- Betriebsmitteldaten (bspw. Rüstzeiten, Lauf- und Stillstandszeiten, Wartezeiten, Maschinenausfallzeiten, Belegungsauswertungen, Gründe für die Stillstandszeiten)
- Personaldaten (bspw. geleistete Arbeitsstunden und Fehlzeiten)
- Qualitätsdaten (bspw. Ausschussstückzahlen und Ausschussgründe)
- Prozessdaten (bspw. technische Daten)

Die BDE lässt sich weiterhin in die *maschinenbezogene Datenerfassung (MDE)* und die *personenbezogene Datenerfassung (PDE)* unterteilen. Zu den Funktionen einer Betriebsdatenerfassung gehört aber nicht nur die Erfassung und Speicherung der Betriebsdaten, sondern auch deren Aufbereitung und Weitergabe, sowie verschiedene Verarbeitungsfunktionen zu deren Korrektur. Dadurch ist es jederzeit möglich, Rückschlüsse auf den Zustand des realen Systems zu ziehen.

Produktionsplanungs- und -steuerungssysteme, Produktionsleitstände, *Enterprise Resource Planning-Systeme (ERP-Systeme)*, Systeme zur Betriebsdatenerfassung oder andere Planungs- und Dokumentationsdatenbanken arbeiten heute überwiegend mit Hilfe relationaler Datenbanken. In der Automobilindustrie wird schon in der Planungsphase auf eine möglichst effiziente und für Auswertungen offene Datenverwaltung geachtet. Aus diesem Grund ist es in den seltensten Fällen notwendig, direkt Daten von Förderbändern oder Maschinen über NC-Steuerungen, speicherprogrammierbare Steuerungen (SPS) oder OPC-Servern zu verwenden. Hierzu existieren bereits Ansätze, um Realdaten von Produktionseinheiten direkt mit Simulationsmodellen zu koppeln ([Fel00]). Da bspw. bei der DaimlerChrysler AG die Daten von BDE- oder PPS-Systemen zeitnah in entsprechenden relationalen Datenbanken gespeichert werden,

beschränkt sich diese Arbeit auf eine Systematik zur Anbindung relationaler Datenbanken an Simulationsbasierte Frühwarnsysteme.

2.2.2 Die Digitale Fabrik

Der letzte Abschnitt hat gezeigt, dass die Durchdringung von Rechnersystemen in der operativen Produktionsplanung und -steuerung sehr hoch ist. Dies geschieht mit Hilfe von PPS-Systemen, Fertigungs- und Montageleitständen, sowie BDE-Systemen. Auch die Produktentwicklung wird durch den Einsatz ausgereifter CAD-Software in großem Ausmaß unterstützt ([Rib00]).

Im Bereich der Planungsdurchführung herrscht überwiegend eine individuelle Datenverarbeitung oder verschiedene Eigenentwicklungen, die zumeist Insellösungen darstellen und keinem übergreifenden Standard entsprechen. Der Planungsprozess ist durchaus beschrieben, doch mangelt es an einer durchgängigen Unterstützung seitens der Informationstechnologie (vgl. Abb. 2.4) ([LGW99]).

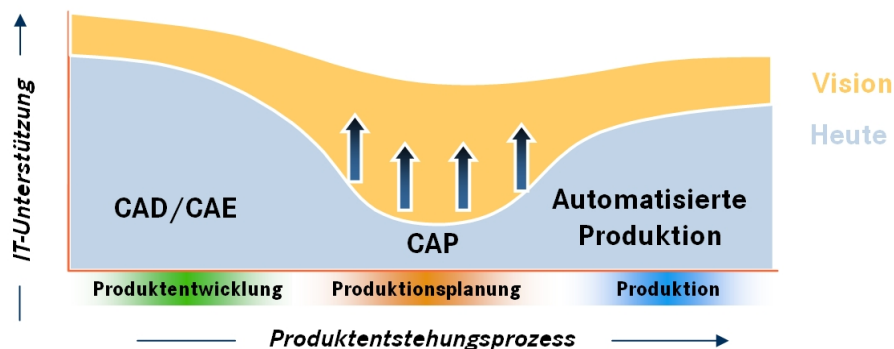


Abb. 2.4: Durchgängige Datenbasis mit Hilfe der digitalen Fabrik ([LGW99])

Mit Hilfe der *Digitalen Fabrik* soll dieser Mangel beseitigt werden. Das Ziel ist eine ganzheitliche Planung, Realisierung, Steuerung und kontinuierliche Verbesserung aller wesentlichen Fabrikprozesse und -ressourcen. Zu diesen Methoden gehört unter anderem die Simulation und die 3D-Visualisierung. Nach der VDI-Richtlinie 4499 wird die Digitale Fabrik wie folgt definiert ([VDI06]):

„Die Digitale Fabrik ist der Oberbegriff für ein umfassendes Netzwerk von digitalen Modellen, Methoden und Werkzeugen [...] die durch ein durchgängiges Datenmanagement integriert werden. Ihr Ziel ist die ganzheitliche Planung, Evaluierung und laufende Verbesserung aller wesentlichen Prozesse und Ressourcen der Fabrik in Verbindung mit einem Produkt.“

Im Gegensatz zu CIM ist die Digitale Fabrik im Gesamtkontext des Unternehmens zu sehen, der Fokus liegt also nicht ausschließlich auf dem Einsatz einer Software. Die Digitale Fabrik lässt sich in die vier Ebenen Datenbasis bzw. Datenkern, Integrationsplattform, Werkzeuge, sowie Organisation und Planungsworkflow untergliedern ([Sch02]). Das Ziel liegt in

der Standardisierung der bei der Planung von Produkten bereits bewährten Methoden, Prozesse und Betriebsmittel zur Weiterverwendung bei nachfolgenden Planungsprojekten. Hierzu ist oftmals eine Überarbeitung der bestehenden Prozesse und der Organisation nötig ([VDI02]). Die Digitale Fabrik ist aber nicht nur auf die Planung beschränkt, sondern erstreckt sich vielmehr über den gesamten Produktentstehungsprozess, von der Entwicklung über die Grob- und Feinplanung bis hin zur Serienplanung und -steuerung.

Die Hauptvorteile der Digitalen Fabrik liegen in einer gemeinsamen Datenbasis und somit einer Reduzierung und Vermeidung redundanter Daten, der Standardisierung von Prozessen, Ressourcen und Betriebsmitteln, einer konsequenten Abklärung von Aufgaben, Kompetenzen und Verantwortung über die Prozesskette in einen gewerkeübergreifenden integrierten Prozess, sowie die Möglichkeiten der Automatisierung ([GB05]).

Die Herausforderungen der Digitalen Fabrik liegen in der Integration in eine bestehende Systemlandschaft und einer mangelnden Leistungsfähigkeit der Planungssoftware, weshalb die Digitale Fabrik noch nicht flächendeckend in der produzierenden Industrie eingesetzt wird. Aufgrund der hohen Kosten und dem unklaren Nutzen wagen sich bisher nur Großunternehmen an die Umsetzung dieser neuen Technologie. Dies führt darüber hinaus zu einem Akzeptanzproblem im operativen Tagesgeschäft und bei der Einführung einer neuen Technologie sind oftmals menschliche Barrieren zu überwinden ([GB05]).

2.2.3 Simulation in Produktion und Logistik

Der Einsatz der Simulation in Produktion und Logistik bezieht sich sowohl auf den gesamten Lebenszyklus technischer Systeme, als auch auf jeden Abschnitt der logistischen Kette eines Unternehmens. Die *VDI-Richtlinie 3633* richtet sich in diesem Zusammenhang an Anwender der Simulationstechnik zur Durchführung von Simulationsstudien von Materialfluss-, Logistik- und Produktionssystemen und sieht die Einsatzmöglichkeiten der Simulation in der Planungs-, Realisierungs- und Betriebsphase technischer Systeme (vgl. Abb. 2.5) ([VDI00]).

Der ursprüngliche Schwerpunkt des Simulationseinsatzes ist der Bereich der Planung und dient sowohl der Planungsabsicherung als auch der Planungsverbesserung. Dabei liegen die Fragestellungen in der Planung meist in der Überprüfung neu geplanter Anlagenkonzepte vor deren Umsetzung, aber auch zur Verbesserung (Redesign, Tuning, Modifikation) vorhandener Anlagen durch Ermittlung von Kapazitätsgrenzen und Schwachstellen ([VDI00]).

Auch im Bereich der Realisierung von technischen Systemen dient die Simulation zur Unterstützung und Ermittlung des Anlaufverhaltens (simulierter Probebetrieb), der Überprüfung von Auswirkungen bei Anforderungsveränderungen, der Erprobung und dem Test von Steuerungssoftware, der verbesserten Mitarbeiterschulung und der Präsentation der funktionalen Übersicht bspw. zum Training von Leitstandspersonal ([VDI00]).

Im Betrieb wird die Simulation sowohl zur Prozesssteuerung als auch zu Parameterstudien eingesetzt. Dabei kommt der Simulation im Bereich der Produktionsplanung und -steuerung eine besondere Bedeutung zu, da die entsprechenden Aufgaben eines PPS-Systems unterstützt werden können ([VDI00]).

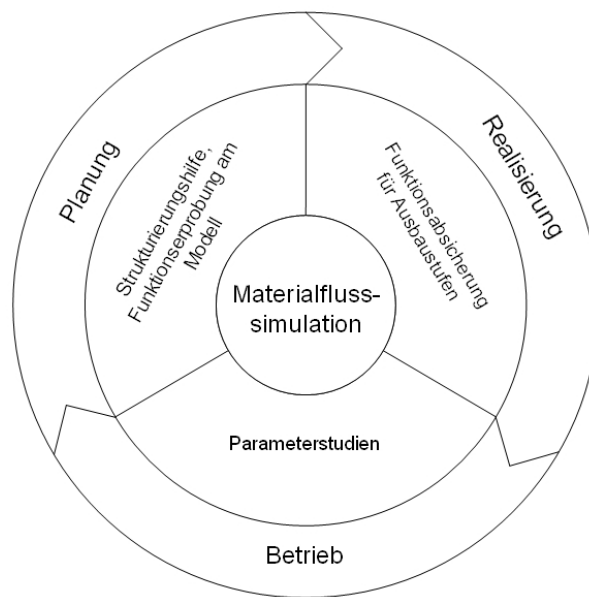


Abb. 2.5: Anwendungsfelder der Simulation ([VDI00])

Nach Bayer wird der Einsatz der Simulation in der Automobilindustrie nicht nach unterschiedlichen Lebenszyklusphasen der technischen Systeme untergliedert, sondern nach deren Funktionalitäten eingeteilt ([Bay03]):

- **Analyse**
bestehender und geplanter technischer Systeme zur Identifikation kritischer Bereiche.
- **Optimierung**
zur Verbesserung bestehender Prozesse und Minimierung der Bestände, sowie Durchführung kostengünstiger „What-if-Analysen“ und Steigerung der Planungsqualität.
- **Schulung**
mittels verständlicher und intuitiver Darstellung mit der Möglichkeit zur Übung und virtuellen Rundgängen.
- **Kontrolle**
zur Ablaufüberwachung im Betrieb.
- **Prognose**
zur Vorhersage von Konfliktsituationen.

In den folgenden Abschnitten wird auf Simulationswerkzeuge zum Einsatz im Bereich von Produktion und Logistik eingegangen, sowie die generelle Vorgehensweise bei der Modellerstellung näher erläutert.

2.2.3.1 Simulationswerkzeuge

Ein Simulationswerkzeug wird für unterschiedliche Zielsetzungen, Anwendungsfelder und Modellierungsphilosophien entwickelt. Grundsätzlich besteht ein Simulator aus den Hauptkomponenten Simulorkern, Datenverwaltung, Benutzeroberfläche und Schnittstellen (vgl. hierzu Abb. 2.6).

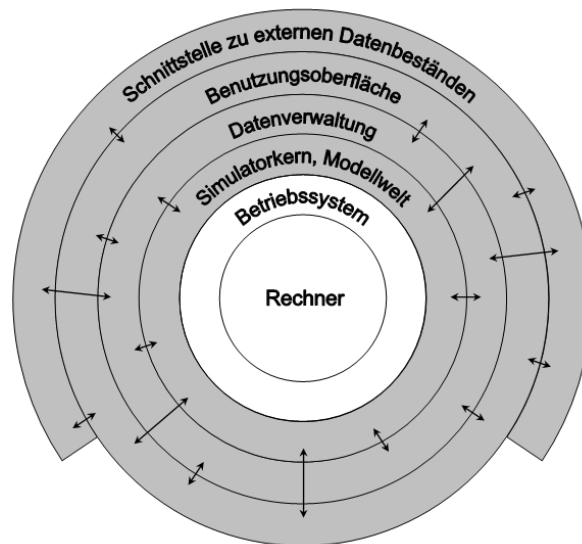


Abb. 2.6: Komponenten eines Simulators ([VDI00])

Dabei stellt der Simulorkern die Modellwelt inklusive ihrer Modellelemente bereit und übernimmt die automatische, chronologische Erzeugung und Verarbeitung von Ereignissen. Er führt die Simulationsberechnungen durch und verknüpft die Komponenten des Simulationssystems. Der Simulorkern stellt die zentrale Ablaufsteuerung eines Simulators dar. Bei den zu verwaltenden Daten handelt es sich um

- **Interne Modelldaten**
Dies sind feste Modellparameter wie bspw. Massen, Trägheitsmomente und Abmessungen.
- **Eingabedaten**
Werden vom Modellnutzer bereitgestellt wie bspw. Störungsdauern.
- **Zustandsdaten**
Verändern sich mit der Modelllaufzeit wie bspw. Geschwindigkeiten oder Beschleunigungen.
- **Resultatdaten**
Werden während und nach dem Simulationslauf ausgegeben, wie bspw. Durchlaufzeiten.
- **Modellbibliotheksverwaltung**
Dient zur Verwaltung von Standard-Modellelementen und Standard-Subsystemen zum Zweck der Wiederverwendung.

Eine Benutzeroberfläche dient zur Editierung und Visualisierung des Simulationsmodells, sowie zur Betrachtung der Simulationsläufe. Weiterhin ermöglicht die Benutzeroberfläche ein Eingreifen während der Simulationsläufe und die Darstellung der aufbereiteten Simulationsergebnisse.

Die meisten Simulationstools bieten Schnittstellen zu Visualisierungstools, Programmen zur Datenauswertung und -darstellung, anderen Simulationssystemen zur Kopplung von Modellen und Simulationsdaten und dem Import von externen Datenquellen wie bspw. Datenbanken oder gar *Excel-Sheets*.

Zur Entwicklung von auf einem Rechner ablauffähigen Simulationsmodellen kommen die folgenden grundsätzlichen Möglichkeiten in Frage:

- **Konventionelle Programmiersprachen**

Das Simulationsmodell wird vollständig in einer konventionellen Programmiersprache wie beispielsweise *C/C++*, *Visual Basic* oder *Java* programmiert. Dadurch ist ein Maximum an Flexibilität möglich, da eine freie Auswahl der Programmiersprache keine lizenzrechtlichen Probleme mit sich bringt. Die Nachteile liegen zum einen in dem erheblichen Implementierungs- und Anpassungsaufwand und zum anderen in der Notwendigkeit fundierter Programmier- und Statistikenntnissen zur Programmierung eines Simulators. Allerdings existieren für einige konventionelle Programmiersprachen bereits umfangreiche Programmbibliotheken wie beispielsweise *J-SIM*, die hierbei unterstützen können ([MSX00]).

- **Simulationsspezifische Sprachen**

Mit Hilfe von speziellen Konstrukten der simulationsspezifischen Sprachen wird die Programmierung von Simulationsmodellen unterstützt. Beispiele für simulationsspezifische Sprachen sind *CSSL*, *DESIRE*, *SLX* oder *MATLAB* ([Beu06]). Die Vorteile liegen in einer hohen Flexibilität und einer zu konventionellen Programmiersprachen vergleichsweise einfachen Implementierung. Allerdings ist auch hier Programmiererfahrung notwendig und darüber hinaus sind Simulationen nur im Rahmen der Entwicklungsumgebung durchführbar.

- **Graphik-orientierte Simulationsumgebungen**

Bei dieser Implementierungsart erfolgt die Modellierung durch eine graphisch-interaktive Modellerstellung und -änderung unter Verwendung abstrakter Modellbausteine. Die Vorteile liegen in einer vergleichsweise aufwandsarmen Erstellung und einfachen Wiederverwendung vorhandener Modelle. Darüber hinaus existiert ein breites Spektrum solcher Werkzeuge auf dem Markt. Die Nachteile liegen in meist sehr hohen Lizenzkosten und wiederum der Einschränkung der Simulation auf die Entwicklungsumgebung.

Bei der Simulation in der Produktion und Logistik in der Automobilindustrie kommen überwiegend graphik-orientierte Simulationsumgebungen zum Einsatz, die bausteinorientierte Konzepte anwenden. Dabei handelt es sich gegenwärtig um Simulationssysteme wie *AutoMod*, *eM-Plant*, *SimPro*, *Witness*, *Arena* und *Quest*.

Diese Simulatoren sind speziell zur Modellierung von Materialflussprozessen entwickelt worden und setzen voraus, dass ein Materialflusssystem aus festen (bspw. Arbeits- und Montagestationen) und beweglichen Einheiten (bspw. Werkstücke, Fördermittel und Kundenaufträge) besteht. Hierzu bieten die Simulatoren vorgefertigte Bausteinkästen mit komplexen Materialflusselementen an ([AF05]). Diese Bausteine können graphisch am Bildschirm dargestellt und parametrisiert werden.

2.2.3.2 Vorgehensweise bei der Modellerstellung

Mit Hilfe des Simulationsexperimentes werden gegebene Szenarien qualifiziert beurteilt, was sonst nur am realen System möglich wäre. Dabei ist der Modellaufbau eine komplexe und kreative Aufgabe, da die Realität durch nachvollziehbare Abstraktionen und Vereinfachungen abgebildet werden muss, ohne an Aussagekraft hinsichtlich des realen Systems zu verlieren ([KR98]). Nach ([VDI00]) ist unter dem Modellbildungsprozess folgendes zu verstehen:

„Die Modellierung umfasst bei der Simulation das Umsetzen eines existierenden oder gedachten Systems in ein experimentierbares Modell.“

Dabei kommt der Bestimmung des Detaillierungsgrades eine große Bedeutung zu. Wird dieser zu gering gewählt, so können die Ergebnisse zu ungenau sein, um diese auf das reale System zu übertragen. Ist der Detaillierungsgrad dagegen zu hoch, so steigt der Modellierungs- und Rechenaufwand erheblich, wodurch aus Zeitgründen eine geringere Anzahl von Experimenten gefahren werden muss und das Ergebnis unter Umständen nicht mehr statistisch gesichert ist ([KR98]). Die Detaillierung darf also nur so exakt wie nötig sein, um die vorgegebenen Untersuchungsziele mit Hilfe des Modells beantworten zu können. Zur Beschreibung des zu untersuchenden realen Systems werden nach VDI 3633

- technische Daten zur Beschreibung der Anlagentopologie, sowie der einzelnen Systemkomponenten,
- Organisationsdaten zur Definition der Arbeitszeit- und Ablauforganisation, sowie der Ressourcenzuordnung und
- Systemlastdaten bestehend aus Auftragseinlastungen und Produktdaten

benötigt ([VDI00]). Diese Eingangsdaten zur Modellierung und der Initialisierung von Simulationsmodellen bei Experimenten beeinflusst die Qualität und Validität des Simulationsmodells. Dabei können die Eingangsdaten reale Ist-Daten oder stochastische Zufallszahlen sein, die Schwankungen unterworfen und somit variabel sind. Unter der Variabilität versteht man dabei die Veränderlichkeit von Ereignissen bezüglich Zeit und Häufigkeit ([Wik07f]).

Nach Aufbau des Modells wird dessen Richtigkeit hinsichtlich seiner Plausibilität verifiziert. Unter der Verifikation wird im Allgemeinen der Vorgang bezeichnet, einen vermuteten oder behaupteten Sachverhalt als wahr zu belegen ([PP04]). Die Ergebnisse der Simulationsläufe bzw. Experimente werden analysiert, also qualitativ und quantitativ ausgewertet.

Die so gewonnenen Erkenntnisse können auf das reale Materialflusssystem übertragen werden. Dieser Vorgang wird auch als Interpretation der Ergebnisse bezeichnet, da die getroffenen Abstraktionen und Vereinfachungen bei der Modellierung berücksichtigt werden müssen. Das Verhalten des realen Systems ist dabei mit dem Simulationsmodell zu vergleichen, um festzustellen, wie genau das Modell das reale System abbildet.

2.3 Entwicklungstendenzen der Simulation in Produktion und Logistik

Die Durchführung von Simulationsstudien zur Entscheidungsunterstützung im Bereich komplexer Produktionssysteme ist sehr zeitaufwändig. Dies beginnt mit der Erfassung von Daten und der Sammlung von Informationen. Die Zielsetzungen der Simulationsstudie sind festzulegen, ein konzeptionelles Modell ist zu entwickeln und es ist eine Projektplanung erforderlich. Das Simulationsmodell muss sowohl erstellt und getestet als auch validiert und verifiziert werden. Anschließend liefert die Durchführung von Simulationsexperimenten Ergebnisse, die zu einer Entscheidungsunterstützung herangezogen werden können ([FR04]). Die Simulation ist eine interdisziplinäre Technologie, die aufgrund ihrer Komplexität ein ausgeprägtes Expertenwissen erfordert (vgl. Abschnitt 2.1.3). Dies und der große Zeitaufwand führen trotz der weiten Verbreitung der Simulation als Standardwerkzeug auch heute noch zu Akzeptanzproblemen im industriellen Umfeld.

In Fowler und Rose werden die künftigen Herausforderungen in der Modellierung und Simulation komplexer Produktionssysteme diskutiert ([FR04]). Hierzu gehören unter anderem die Reduzierung des zeitlichen Aufwandes bei der Durchführung von Simulationsstudien. Zum einen muss die Zeit zur Informations- und Datenbeschaffung verringert werden. Dies kann durch eine effiziente und automatisierte Datenaufbereitung, der Verbesserung der Datenqualität im industriellen Umfeld und durch die Verwendung standardisierter Datenaustauschformate geschehen. Darüber hinaus sind auch Reduzierungen des Aufwandes beim Modelldesign, bei der Modellentwicklung und dem Modelleinsatz möglich. Die Verkürzung von Simulationsexperimenten kann bspw. durch den konsequenten Einsatz der verteilten und parallelen Simulation erreicht werden.

Eine weitere Tendenz ist die vermehrte Verwendung der Simulation als eingebettete Applikation in Produktionssteuerungs-, Decision-Support- oder Scheduling-Systemen ([Bru00]). Darüber hinaus gibt es Bestrebungen, die Simulation als Dienstleistung über das Web zur Verfügung zu stellen. In diesem Zusammenhang ist auch die Web-basierte Simulation zu nennen, worunter im engeren Sinne die Ausführung von Simulationsmodellen auf einem Webserver oder deren Ausführung als Applet in einem Web-Browser verstanden werden kann. Die Anwender solcher Systeme müssen keine Simulationsexperten sein, da sie keine direkten Berührungspunkte mit der Simulation haben.

Weitere Erleichterungen für den Simulationsanwender sehen Fowler und Rose in der Entwicklung von Methoden zur schnellen Bereitstellung von Simulationsressourcen ([FR04]). Dies sind beispielsweise Simulationsmodelle der realen Produktion, die permanent zu aktualisieren und zu synchronisieren sind. Wird ein weiteres Simulationsmodell zur Entscheidungsunterstützung benötigt, so muss lediglich ein Klon von diesem Ursprungsmodell erzeugt werden. Eine

andere Möglichkeit besteht in der automatischen Modellgenerierung, die bei Bedarf Simulationsmodelle aus den Fabrikdatenbanken erzeugt und anschließend von einem Simulationsanwender zu Experimenten herangezogen werden kann. Die schnelle und einfache Verknüpfung von Simulationsmodellen und anderen Support-Systemen wie bspw. Produktionsleitständen, Analysewerkzeugen oder Schedulingssystemen ist eine weitere künftige Herausforderung. Einzelne Komponenten und Anwendungen sollen auf einfache Art und Weise für eine bestimmte Aufgabenstellung miteinander interagieren.

In diesem Zusammenhang sind auch Simulationsbasierte Frühwarnsysteme zu sehen. Die Simulation wird als eingebettete Anwendung zur Entscheidungsunterstützung verwendet. Hierzu sind Simulationsmodelle erforderlich, die ein reales Produktionssystem abbilden und eine Frühwarnung vor unerwünschten Ereignissen ermöglichen. Diese Modelle müssen mit den realen Systemzuständen versorgt werden. Weiterhin sind mehrere Funktionskomponenten und Anwendungen in einer offenen und erweiterbaren Architektur miteinander zu verknüpfen, die sich harmonisch in die bestehende IT-Infrastruktur eines Unternehmens einfügen lässt.

Im Zusammenhang mit einer solchen Architektur sind noch ungelöste Fragestellungen zu klären.

Hierzu ist zu diskutieren, welche Komponenten bzw. Funktionalitäten in einem Simulationsbasierten Frühwarnsystem benötigt werden, um zum einen die Simulation als eingebettete Applikation zu verwenden und zum anderen die effiziente Integration in eine bestehende Systemwelt zu ermöglichen. Dabei stellt sich die Frage, wie die Komplexität der Simulation und Integration der einzelnen Funktionskomponenten vor dem Anwender verborgen werden kann und wie eine generelle Architektur Simulationsbasierter Frühwarnsysteme aufgebaut werden muss.

Weiterhin ist die Synchronisation der verwendeten Simulationsmodelle bzw. des Simulationsbasierten Frühwarnsystems mit dem Zustand des realen Systems von Interesse. Dabei ist zu klären, was unter einer Frühwarnung zu verstehen ist bzw. wie zu detektierende Ausnahmefälle dargestellt und beschrieben werden können, bei deren realen oder zukünftigen Auftreten eine Warnung erwünscht ist. Hierbei spielt die Simulation zur Prognose künftiger Zustände des realen Systems eine entscheidende Rolle.

Ein weiteres zu diskutierendes Themengebiet ist die flexibel und effizient zu gestaltende Datenerfassung und die Anbindung Simulationsbasierter Frühwarnsysteme an Fabrikdatenbanken und Produktivsysteme. Dabei ist ein verstärktes Augenmerk auf eine Standardisierung des Datenaustausches zu legen. Darüber hinaus ergeben sich Fragestellungen zur Gestaltung der Entscheidungsunterstützung bei erfolgter Frühwarnung. Hier ist zu klären, wie mit Ausnahmefällen umzugehen ist und wie eine Lösungsstrategie aussieht, um mit diesen in geeigneter Weise zu verfahren.

Abschließend sind reale Anwendungsfälle zu diskutieren, die sowohl die An- und Umsetzbarkeit als auch die Potenziale Simulationsbasierter Frühwarnsysteme aufzeigen.

2.4 Zusammenfassung des Kapitels

In diesem Kapitel wird nach einer allgemeinen Einführung in das Themengebiet der Simulation auf die Computersimulation eingegangen. Hierzu erfolgt eine kurze Beschreibung der geschichtlichen Entwicklung der Computersimulation bis hin zum heutigen Standardwerkzeug für Ingenieure und Wissenschaftler. Nach der Vorstellung grundsätzlicher Rahmenbedingungen für einen sinnvollen Einsatz der Simulation, werden mit Hilfe von Definitionen die Begriffe Simulation, Simulationsmodell, Simulator und Simulationsexperiment erläutert. Anschließend wird die Simulation nach den verwendeten Eingangsgrößen, der Dynamik der Modelle und der Art der Zeitsteuerung klassifiziert. Dabei sind vor allen Dingen die grundsätzlichen Methoden der diskreten Simulation interessant. Das Themengebiet der Computersimulation wird mit einem Abschnitt zu deren Anwendungsbereichen, Nutzen und Grenzen abgeschlossen.

Im Anschluß daran wird die Simulation im Kontext der Automobilindustrie betrachtet. Hier liegen die Haupteinsatzgebiete sowohl in der Planung von Produktionsanlagen, Logistikbereichen und -konzepten als auch in der operativen Produktionsplanung und -steuerung. Dabei werden die zunehmenden Herausforderungen in der Automobilindustrie und der gerechtfertigte Einsatz der Simulation diskutiert. Ein weiterer Abschnitt beschäftigt sich mit der informationstechnischen Integration der Simulation in einem industriellen Umfeld, da diese erhebliche Datenmengen benötigt. Hierbei sind besonders die Themengebiete Computer Integrated Manufacturing (CIM), Produktionsplanungs- und Produktionssteuerungssysteme (PPS) und die Betriebsdatenerfassung (BDE) von Interesse. Es folgen Erläuterungen zur Digitalen Fabrik und deren Bedeutung. Anschließend wird speziell die Simulation in Produktion und Logistik betrachtet, wobei deren Einsatzgebiete vor allem in der Analyse, Optimierung, Schulung, Kontrolle und Prognose liegen. Auf das Themengebiet der Simulationswerkzeuge wird ebenso eingegangen wie auf die Vorgehensweise bei der Modellerstellung.

Zum Abschluss dieses Kapitels werden Entwicklungstendenzen der Simulation in Produktion und Logistik diskutiert, die auch im Zusammenhang mit Simulationsbasierten Frühwarnsystemen von Interesse sind. Hierbei sind bestimmte Schwerpunkte zu betrachten, die im bei der Umsetzung einer offenen und erweiterbaren Architektur berücksichtigt werden müssen.

Kapitel 3

Simulationsbasierte Frühwarnsysteme

*Das Schicksal setzt nie einen König matt,
ehe es ihm Schach geboten hat.*
— Ludwig Börne

Dieses Kapitel beschäftigt sich mit Simulationsbasierten Frühwarnsystemen. Dabei wird zunächst auf Frühwarnsysteme im Allgemeinen und auf deren grundlegende Funktionalitäten eingegangen. In einem weiteren Abschnitt werden Simulationsbasierte Frühwarnsysteme zum Einsatz in Produktion und Logistik diskutiert. Hierzu erfolgt sowohl eine Definition und die Formulierung von Anforderungen an Simulationsbasierte Frühwarnsysteme als auch die Abgrenzung von bestehenden Forschungs- und Themengebieten. Der darauf folgende Abschnitt stellt eine allgemeine Architektur mit deren notwendigen Komponenten vor. In diesem Zusammenhang werden verteilte Systeme und geeignete Kommunikationsprinzipien betrachtet. Zum Abschluss fasst sich ein Abschnitt mit der Steuerung des notwendigen Workflows Simulationsbasierter Frühwarnsysteme.

3.1 Frühwarnsysteme

Frühwarnsysteme finden sich in vielfältigen Einsatzgebieten. So werden Frühwarnsysteme für Naturkatastrophen wie bspw. Tsunamis, Vulkanausbrüche und Erdbeben oder Stürme und Flutkatastrophen angewendet. Auch im sozialen und ökologischen Umfeld sind Frühwarnsysteme zu finden. Hierzu gehören sowohl Frühwarnsysteme gegen humanitäre Katastrophen wie Hungersnöte und Epidemien ([BSDL91],[Dor06]) als auch Flächenbrand-Frühwarnsysteme. Im Bereich von Fahrzeugen und technischen Anlagen dienen Frühwarnsysteme zur rechtzeitigen Abwendung von Gefahren für Personen, die diese Fahrzeuge und Anlagen bedienen und benutzen. Zu strategischen Frühwarnsystemen gehören sowohl Früherkennungssysteme zur Vermeidung möglicher Fehlentwicklungen in Unternehmen als auch militärische Frühwarnsysteme.

Im Allgemeinen sind Frühwarnsysteme

„[...] Einrichtungen, welche aufkommende Gefahren frühzeitig als solche erkennen und potentiell Gefährdete möglichst schnell darüber informieren, um durch rechtzeitige Reaktion die Gefahr abzuwenden oder zu mildern.“ ([Wik07c])

Dabei lassen sich grundsätzlich drei Phasen bei einer Frühwarnung unterscheiden. Die *Vorhersage*, die *Warnung* und die *Reaktion* ([Kat01],[Wik07b]).

Klassische Frühwarnsysteme basieren auf historischen und aktuellen Messwerten von Frühindikatoren zur Bewertung und Erkennung kritischer Systemzustände und potentieller Bedrohungen. Ein Frühwarnsystem besteht aus folgenden grundlegenden Funktionalitäten ([oEW06],[Wik07c]):

- **Messung**
An einem oder mehreren Messpunkten werden Messdaten erhoben. Dies geschieht manuell oder automatisch mit Hilfe verschiedener Sensoren oder anderer technischer Messvorrichtungen.
- **Datensammlung**
Die an den örtlich verteilten Messpunkten erfassten Messdaten werden aggregiert und zur späteren Auswertung gesammelt.
- **Überwachung**
Die permanente Durchführung von Messungen und die Datensammlung dient der Überwachung eines Systems, um Unregelmäßigkeiten und Besonderheiten festzustellen. Hierbei werden oftmals historische Daten zu Vergleichszwecken herangezogen und daraus mit Schwellen- oder Maximalwerten verknüpfte Frühwarnindikatoren oder Regeln entwickelt. Eine Besonderheit bzw. Unregelmäßigkeit entspricht dann einer Abweichung von einer Regel oder der Überschreitung bestimmter Schwellen- oder Maximalwerte.
- **Einschätzung**
Bei dieser Komponente spielt der Faktor Mensch eine große Rolle, da bei den meisten Frühwarnsystemen ein Expertenteam die ermittelten Indizien auf ihre Bedeutsamkeit hin überprüft. Dabei wird die Eintrittswahrscheinlichkeit eines Unglückes abgewogen, wobei eventuell verwendete Regeln oder Frühwarnindikatoren und das Verhältnis der Schäden durch das Unglück und der Schäden durch einen Fehllalarm mit einbezogen werden. Diese Instanz entscheidet schließlich, ob eine Warnung ausgegeben wird oder nicht.
- **Warnung / Verbreitung**
Wird die Entscheidung gefällt, eine Warnung auszugeben, so muss diese so schnell wie möglich verbreitet werden. Hierzu existieren Alarmierungspläne, die eine genaue Reihenfolge der zu alarmierenden Personen oder Einrichtungen festlegt.
- **Automatische Reaktion**
Die Reaktionen auf die Herausgabe von Warnungen müssen nicht an den Menschen gekoppelt sein. Es können auch Automatismen existieren, die ohne menschliche Einwirkung funktionieren (bspw. die automatische Versiegelung von Brandschutztüren oder die Verriegelung von Gasleitungsventilen).
- **Entwarnung**
Falls die Gefahr vorüber ist, werden gefährdete Personen darüber informiert. Dabei wird

oftmals schon bei der Herausgabe einer Warnung auf eine mögliche Entwarnung hingewiesen. Dies verhindert die Fortsetzung von unnötigen Gegenmaßnahmen, obwohl die Gefahr bereits vorüber ist.

- **Voraussetzungen / Vorsorge**

Ein Frühwarnsystem arbeitet nur dann effizient, wenn die Gefährdeten mit der Warnung etwas anfangen können. Aus diesem Grund werden groß angelegte Übungen durchgeführt, die im Ernstfall eine passende Reaktion ermöglichen. Hierzu gehört aber auch bspw. das Anlegen von Schutzräumen, Fluchtwegen oder Vorratsdepots für den Notfall.

Die Abwesenheit einer oder mehrerer dieser Komponenten ist abhängig von der Zielsetzung und dem Einsatzgebiet.

Im Allgemeinen sind Frühwarnsysteme ein wichtiger Bestandteil des Krisen- und Notfallmanagements und kommen in Lebensbereichen vor, in denen mit hoher Wahrscheinlichkeit Gefahren für Personen und Einrichtungen auftreten. Eine erfolgreiche Frühwarnung ist eine rechtzeitige Warnung vor drohenden Ereignissen, die potentiell Betroffenen die Möglichkeit zur Reaktion und zur Vermeidung oder Verminderung von Personen- bzw. Sachschäden gibt. Ein Frühwarnsystem dient der Vorsorge, die sich aus den Teilbereichen *Risikoanalyse*, *Vorbeugung* und *Vorbereitung* auf den Ernstfall zusammensetzt. Hierzu gehören bspw. Massnahmen wie die Etablierung von Einsatz- und Koordinationsstrukturen, Erstellung von Notfallplänen, Durchführung von Notfallübungen und die Organisation einer rechtzeitigen Weiterleitung von Informationen an gefährdete Personen.

Strategische Frühwarnsysteme in Unternehmen stützen sich auf die Verwendung von Frühindikatoren, die mit Sollwerten und Toleranzgrenzen kombiniert werden. Diese Frühindikatoren werden in Abhängigkeit der gewünschten Beobachtungsbereiche entwickelt und sind mit einer effizienten Informationsverarbeitung zu versehen ([Rea05]). Bei Überschreitung von Schwellenwerten für festgelegte Größen wird eine Frühwarnung ausgelöst. In einem Unternehmen kommen für solche Größen Leistungsmaßstäbe wie der *Cash Flow*, *Zinsfaktoren*, *Inflation*, *Eigenkapitalrendite*, *Kapitalrendite*, *Umsatzrendite* oder der *Return on Investment (ROI)* in Frage.

Die Problematik hierbei ist die fehlende Abbildung künftiger Ereignisse, da die Frühwarnung ausschließlich auf Vergangenheitsdaten beruht. Aus diesem Grund werden Frühindikatoren verwendet, die mit Hilfe der Risikoidentifikation, der Risikoanalyse, der Risikobewertung und der Risikoaggregation ausgearbeitet werden ([Axe03]). Sowohl unternehmensinterne als auch -externe relevante Ereignisse und Entwicklungen, aus denen sich Chancen und Risiken für das Unternehmen ergeben können, werden systematisch überwacht. Diese Methode berücksichtigt allerdings nur quantitative Risiken, nicht jedoch qualitative Veränderungen wie etwa Fluktuationen oder Fehlentscheidungen im Management.

In nachfolgendem Abschnitt wird auf Simulationsbasierte Frühwarnsysteme für den Einsatz in Produktion und Logistik eingegangen.

3.2 Simulationsbasierte Frühwarnsysteme in Produktion und Logistik

In diesem Abschnitt wird die Thematik Simulationsbasierter Frühwarnsysteme zum Einsatz in der Produktion und der Logistik diskutiert. Es existiert noch keine geeignete und umfassende Definition Simulationsbasierter Frühwarnsysteme und deren Abgrenzung zu anderen simulationsbasierten Ansätzen in der Literatur. Weiterhin sind Anforderungen zu formulieren, die einen effizienten Einsatz in Produktion und Logistik ermöglichen. Dies umfasst den gesamten innerbetrieblichen Materialfluss in einem Unternehmen, der zur Beförderung von Gütern

- zwischen einzelnen betrieblichen Bereichen,
- innerhalb eines Bereiches oder
- von einem Arbeitsplatz zu einem anderen Arbeitsplatz

dient ([AF05]). Dabei sind diese Bereiche und Arbeitsplätze in einer Vorgangsfolge nach technischen und organisatorischen Regeln miteinander verknüpft und bilden ein Logistik- bzw. Produktionssystem. In diesem System werden Roh-, Hilfs- und Betriebsstoffe, angearbeitete und fertige Teile, Baugruppen und Produkte aller Art befördert, für die der Oberbegriff *Material* verwendet werden kann. In diesem Zusammenhang soll im weiteren Verlauf von Materialflusssystemen gesprochen werden.

Die Globalisierung bewirkt einen Wandel der Technologien und eine erhöhte Mobilität der Produktionsfaktoren. Dies führt zu zunehmender Wettbewerbsintensität, Transparenz der Märkte und weltweiten Überkapazitäten. Produzierende Unternehmen reagieren auf diese Herausforderungen mit kürzeren Produktlebenszyklen, verstärkter Kundenorientierung, einer Individualisierung von Produktion und Leistung und somit einem größeren Produktportfolio ([Die01]). Es findet ein zeitlicher Wettlauf um Produkt- und Prozessinnovationen statt, wobei sich diese Entwicklungen in Form höherer Anforderungen in der Produktentstehungs- und Produktionsplanung bemerkbar machen.

Eine große Variantenvielfalt und die verstärkte Kundenorientierung bewirken eine zunehmende Komplexität und Dynamik in der Produktion. Dies führt zu steigenden Anforderungen und einer zunehmenden Unsicherheit in der Produktionssteuerung bei steigendem Kostendruck. Diesem Kostendruck soll bspw. durch möglichst geringe Bestände und einer gesteigerten Flexibilität der Produktionssysteme entgegen getreten werden. Die Forderung nach geringen Beständen setzt eine schnelle Reaktionszeit der Systeme voraus, da sich Störgrößen wie beispielsweise Maschinenausfälle oder kurzfristige Veränderungen des Produktionsprogramms sehr viel stärker auswirken und die Prozesssicherheit eines Materialflusssystems gefährden. An dieser Stelle können Simulationsbasierte Frühwarnsysteme helfen, da die Vorhersage von Ereignissen, Zuständen oder Entwicklungen einen zeitlichen Vorteil ermöglicht. Es können frühzeitig geeignete Reaktionen oder Gegenmaßnahmen erfolgen.

Bei der Steuerung eines Materialflusssystems sind ständig kurzfristige Entscheidungen zu treffen. Dabei beinhaltet die Reaktivität Handlungen, die auf vorangegangene Aktionen oder den

aktuellen Systemzustand angepasst sind. Dagegen berücksichtigt proaktives Verhalten ein vorausschauendes und zielgerichtetes Handeln als Reaktion auf mögliche oder reale Zustandsänderungen.

Klassische Frühwarnsysteme beruhen ausschließlich auf historischen und aktuellen Messdaten von Frühindikatoren, kritischen Systemzuständen und potentiellen Bedrohungen. Die zu Beginn des Abschnittes 3.1 dargelegten Funktionalitäten eines Frühwarnsystems sind auch für Simulationsbasierte Frühwarnsysteme in den Bereichen Produktion und Logistik weitestgehend gültig. Im Unterschied zu klassischen Frühwarnsystemen verwenden Simulationsbasierte Frühwarnsysteme die Simulation als integriertes Prognosetool.

Einige Frühwarnsysteme zur Wettervorhersage oder für Naturkatastrophen nutzen bereits Simulationsmodelle zur Prognose eventueller Gefahren oder zur Beurteilung möglicher Auswirkungen bestimmter Ereignisse. Aufgrund der Zunahme der Komplexität realer Produktionssysteme ist die Simulation besonders zur Prognose künftiger Systemzustände geeignet (vgl. Rahmenbedingungen für einen sinnvollen Einsatz der Simulation in Abschnitt 2.1). Darüber hinaus stellt die Prognosefunktionalität eine Eigenschaft und ein Ziel der Simulation dar (vgl. Abschnitt 2.2.3).

Die klassische Anwendung der Simulation in Verbindung mit den Bereichen Produktion und Logistik liegt in der Unterstützung der Planung, Parametrierung und Optimierung mit dem Ziel, grundlegende Erkenntnisse über ein Materialflusssystem zu erhalten ([KR98]). Soll ein Materialflusssystem mit einem Simulationsbasierten Frühwarnsystem ausgestattet werden, wird ein Simulationsmodell des realen Systems benötigt. Mit Hilfe dieses Modells ist es möglich, noch vor dem Eintritt einer Ausnahmesituationen diese zu erkennen und geeignete Gegenmaßnahmen zu ergreifen. Darüber hinaus können mit Hilfe der Simulation mehrere alternative Gegenmaßnahmen (im Folgenden als Handlungsalternativen bezeichnet) bewertet werden.

Es ist zu klären, aus welchen Komponenten ein Simulationsbasiertes Frühwarnsystem besteht und wie diese Komponenten miteinander in Beziehung stehen. Weiterhin stellt sich die Frage, wie eine geeignete Steuerung des Frühwarnsystems arbeitet und welche Anforderungen an Simulationsbasierte Frühwarnsysteme zu stellen sind.

Bevor diese Fragestellungen in den nachfolgenden Abschnitten beantwortet werden, erfolgt zunächst die Herleitung einer allgemeingültigen Definition Simulationsbasierter Frühwarnsysteme.

3.2.1 Eine Definition

In Abschnitt 3.1 wird die Definition eines Frühwarnsystems bereits vorgestellt. Im Allgemeinen soll ein Frühwarnsystem aufkommende Gefahren rechtzeitig erkennen. Damit ist der Sinn eines Frühwarnsystems allerdings noch nicht erfüllt. Es muss weiterhin gefährdete Personen möglichst schnell darüber informieren, damit frühzeitig Gegenmaßnahmen ergriffen werden

können. Klassische Frühwarnsysteme basieren ausschließlich auf Vergangenheits- und aktuellen Meßdaten. Simulationsbasierte Frühwarnsysteme, die im Folgenden als *SEWS (Simulation-based Early Warning Systems)* bezeichnet werden, verwenden die Simulation zur Vorhersage künftiger Systemzustände.

Ein weiterer Unterschied liegt in der Bewertung von Handlungsalternativen bzw. proaktiven Steuerungsmaßnahmen. Dies ist keine notwendige Funktionalität eines Frühwarnsystems, da die Hauptaufgabe in der frühzeitigen Erkennung von Gefahren und der Warnung von Gefährdeten liegt. Für den Einsatz solcher Systeme (beispielsweise in der Automobilindustrie) ist es allerdings sehr wichtig, zusammen mit der Warnung bereits sinnvolle und umsetzbare Handlungsalternativen vorzuschlagen. Dies ist zum einen für die Akzeptanz von SEWS von Bedeutung und zum anderen aufgrund wirtschaftlicher Aspekte notwendig. Verschiedene Handlungsalternativen müssen nicht manuell entwickelt und miteinander verglichen werden. Dies kann unter Umständen sehr zeitraubend sein und zu ungeeigneten Ergebnissen führen.

Berücksichtigt man alle diese Komponenten, ergibt sich die Definition Simulationsbasierter Frühwarnsysteme in folgender Weise:

„Ein Simulationsbasiertes Frühwarnsystem ist eine Einrichtung, welche basierend auf einem Simulationsmodell des entsprechenden komplexen realen Systems so schnell wie möglich auf negative Effekte oder positive Potenziale durch Prognose und Beurteilung unterschiedlicher Handlungsalternativen hinweist und diese dem Nutzer des Gesamtsystems in geeigneter Form mitteilt.“

Es stellt sich nun die Frage, was unter negativen Effekten oder positiven Potenzialen zu verstehen ist. In dieser Arbeit geht es um SEWS zum Einsatz in der Produktion und Logistik. Ein in einem realen System auftretender negativer Effekt ist im Grunde genommen die negative Abweichung von einer oder mehreren für das reale System bestehenden Regeln. Der Ausdruck *negativ* deutet darauf hin, dass diese Abweichung eine nicht gewünschte Richtung von einem gewünschten Zustand einschlägt. Bei Produktionssystemen sind dies beispielsweise Abweichungen vom geplanten Produktionsvolumen. Dabei kann sowohl eine zu hohe als auch eine zu geringe Produktion negativ sein. In dem einen Fall könnten dadurch kostenwirksame Bestände aufgebaut werden, in dem anderen Fall werden Produktionsaufträge nicht rechtzeitig erfüllt.

Potenziale sind die in einem System innewohnende Fähigkeit oder Funktionalität zur Bewältigung einer Aufgabe ([Wik07e]). In diesem Fall ist mit einem positiven Potenzial die Möglichkeit gemeint, von den bestehenden Regeln eines Systems abzuweichen und dadurch eine Verbesserung der künftigen Situation zu erreichen. Ein Beispiel in der Produktion und Logistik hierfür wäre die Feststellung, dass ein gewünschtes Produktionsprogramm mit einem geringeren als dem geplanten Ressourceneinsatz gestaltet werden kann.

Eine Ausnahme ist die Abweichung von einer definierten oder gegebenen Regel ([Bro02]). Aus diesem Grund soll das Auftreten negativer Effekte oder positiver Potenziale im Folgenden als Ausnahmefall bezeichnet werden.

Ein effizienter Einsatz von SEWS im Bereich der Produktion und Logistik stellt hohe Anforderungen an deren Design und Architektur, die bei deren Entwicklung zu berücksichtigen sind. Diese Anforderungen werden im folgenden Abschnitt erläutert.

3.2.2 Anforderungen an Simulationsbasierte Frühwarnsysteme

Die industrielle Anwendung von SEWS stellt hohe Ansprüche an deren Fähigkeiten. Solche Systeme müssen schnell reagieren können und flexibel auf vielfältige Problemstellungen anwendbar sein. Ausnahmefälle sind sowohl im aktuellen Systemzustand als auch in den mit Hilfe der Prognose ermittelten zukünftigen Systemzustände zu erfassen. Im Ausnahmefall müssen möglichst sinnvolle Handlungsalternativen in kurzer Zeit zur Verfügung stehen, die in der betrieblichen Praxis umsetzbar sind. Weiterhin ist ein stabiler und kostengünstiger Betrieb eines SEWS zu ermöglichen.

All dies führt zu folgenden allgemeinen Anforderungen, die bei dem Design, der Entwicklung und dem Betrieb von SEWS berücksichtigt werden müssen:

- **Performance**

Die Geschwindigkeit, mit der ein SEWS Prognosen erstellt und Handlungsalternativen vorschlägt ist von großer Bedeutung. Je höher die Geschwindigkeit, mit der dies geschieht, desto größer ist die Akzeptanz bei den Anwendern. Die Reaktionszeit eines SEWS wird entscheidend von der Performance der verwendeten Simulatoren, des „*Level of Detail (LOD)*“ der Simulationsmodelle, der Verarbeitungsgeschwindigkeit der Daten und der Effizienz der Koordination der beteiligten Komponenten beeinflusst.

- **Reaktives Verhalten**

Über- oder unterschreiten bestimmte Werte des aktuellen Systemzustandes den Schwellenwert eines definierten Ausnahmefalls, so generiert das SEWS Handlungsalternativen, die mit Hilfe der Simulation zu bewerten sind. Aus diesen Handlungsalternativen können eine oder mehrere ausgewählt werden, um die Auswirkungen einer Ausnahmesituation zu vermindern. Hierzu können unterschiedliche Kriterien wie Kosten, Anzahl eingesetzter Mitarbeiter oder Bestandsentwicklungen verwendet werden. Dies entspricht der Zielsetzung von Monitoringsystemen ([Fel00]). Darüber hinaus können benutzerdefinierte Ausnahmefälle herangezogen werden, um künftige geplante oder bereits bekannte Ausnahmefälle (bspw. die Auswirkungen von Maschinenausfällen oder unterschiedlichen Steuerungsprinzipien) auf Basis des aktuellen Systemzustandes zu bewerten. Solche Ausnahmefälle werden von einem Anwender des SEWS definiert und werden im Folgenden als *User Exception* bezeichnet.

- **Proaktives Verhalten**

Mit Hilfe der Simulation werden kritische Ausnahmefälle auf Basis des aktuellen Systemzustandes prognostiziert. Dadurch ist es möglich, schon vor Eintritt einer Ausnahmesituation geeignete Gegenmaßnahmen einzuleiten, die in Form von Handlungsalternativen von einem SEWS vorgeschlagen werden. Diese verhindern im Idealfall die Überschreitung eines Schwellenwertes oder verringern den Grad der Überschreitung.

- **Skalierbarkeit / Adaptionfähigkeit**

Unter der Skalierbarkeit versteht man die Anpassungsfähigkeit eines Systems an geänderte, meist gesteigerte Anforderungen bei laufendem Betrieb ([GB03]). Ein SEWS muss in der Lage sein, durch autonome Modifikation von Modellparametern auf Abweichungen der Prognosen von tatsächlichen Verläufen zu reagieren. Dadurch nimmt die Abbildungsgenauigkeit der Modelle zu und die Prognosegenauigkeit wird verbessert. Die Simulationsmodelle müssen sich durch den Abgleich mit den vorhandenen Datengrundlagen validieren und hinsichtlich wichtiger Kennwerte aktuell halten. Weiterhin ist die Definition unterschiedlichster Ausnahmefälle mit Hilfe einer offenen Datenstruktur zu unterstützen.

- **Erweiterbarkeit**

Eine Erweiterung des zu unterstützenden realen Systems um weitere Parameter oder Systembereiche muss in einfacher Weise und mit geringem Aufwand möglich sein und sich auch in den Simulationsmodellen widerspiegeln. Dadurch ergeben sich hohe Anforderungen hinsichtlich Offenheit und Verwendung von Standards bei der Umsetzung einer SEWS-Architektur.

- **Unabhängigkeit**

Die Anforderung der Unabhängigkeit ergibt sich aus der gewünschten Skalierbarkeit bzw. Adaptionfähigkeit und Erweiterbarkeit. Ein SEWS muss auf unterschiedlichen Betriebssystemen lauffähig und somit plattformunabhängig sein. Weiterhin müssen unterschiedliche existierende Simulationsmodelle bzw. Simulatoren integriert werden können. Die benötigten Daten werden in heterogenen Datenquellen gespeichert bzw. die aktuellen Prozessdaten werden über entsprechende Systeme online eingebunden. Auch hier muss ein SEWS weitestgehend unabhängig von den Datenstrukturen bestehender Systeme sein.

- **Benutzerfreundlichkeit**

Ein SEWS muss sich durch eine hohe Benutzerfreundlichkeit auszeichnen. Die Nutzung soll keinem eingeschränkten Benutzerkreis mit hohem Schulungsaufwand vorbehalten sein. Daraus ergeben sich Anforderungen wie bspw. die Konfigurierbarkeit verschiedener Benutzersichten, verschiedene Alarmierungsschemata oder eine Benutzergruppenverwaltung.

- **Lernfähigkeit**

Um Handlungsalternativen vorschlagen zu können, ist eine gewisse Intelligenz des Systems erforderlich. Geeignete Verfahren zum Wissenserwerb und zur Wissenserweiterung müssen vorhanden sein, um zum einen sinnvolle Alternativen zu generieren und zum anderen unsinnige oder nicht relevante Alternativen aus Performancegründen bereits frühzeitig auch ohne den Einsatz der Simulation auszuschließen.

- **Standardisierung / Wiederverwendbarkeit**

Bei der Entwicklung und Etablierung von SEWS ist auf die Standardisierung und Wiederverwendbarkeit des Gesamtsystems und seiner Komponenten zu achten. Dies kann durch die Anwendung von XML als standardisiertes Datenaustauschformat garantiert werden. Die Wiederverwendbarkeit und einfache Ersetzbarkeit von Komponenten lässt

sich durch die Anwendung von Web Services erreichen. Ein Web Service ist vereinfachend ein Dienst, der über das Internet bzw. Intranet aufgerufen werden kann und in XML beschrieben wird ([GSB⁺02]).

- **Automatismus**

Ein SEWS muss aus Performancegründen und aufgrund der Fehleranfälligkeit eventueller manueller Vorgänge so weit wie möglich ohne menschliches Eingreifen stabil arbeiten. Weiterhin soll die Prognose von Ausnahmefällen, die Generierung von Handlungsalternativen und die Datenbeschaffung automatisch geschehen. Lediglich die Reaktion auf Ausnahmefälle durch die reale Ausführung einer oder mehrerer Handlungsalternativen bleibt dem Anwender eines SEWS überlassen.

Nicht alle diese Anforderungen können mit Hilfe der in Abschnitt 3.1 beschriebenen Komponenten eines Frühwarnsystems beeinflusst werden. Dabei handelt es sich um die Skalierbarkeit bzw. Adaptionfähigkeit, die Erweiterbarkeit, die Unabhängigkeit und die Standardisierung bzw. Wiederverwendbarkeit. Diese Anforderungen sind spezifisch für Frühwarnsysteme mit informationstechnologischem Hintergrund, die nicht bedingungslos auf allgemeine Frühwarnsysteme angewendet werden können. Da der Einsatz von SEWS in der Produktion und Logistik sehr stark von der Anwesenheit geeigneter Informations- und Rechnersysteme abhängig ist, sind diese zusätzlichen Anforderungen bei deren Entwurf zu berücksichtigen.

Im folgenden Abschnitt werden SEWS von anderen Forschungs- und Themengebieten abgegrenzt. Hierzu wird zunächst auf den Begriff der *Online-Simulation* eingegangen. Anschließend erfolgt eine Betrachtung der Bereiche *Real-Time-Simulation*, *Monitoringsysteme*, *Simulationsbasierte Leitstände*, *Betreibersimulation*, *Hardware-In-The-Loop*- und *Real-Time-Simulation* im Zusammenhang mit SEWS.

3.2.3 Abgrenzung der Problemstellung

In der Literatur wird der Begriff *Online-Simulation* sehr häufig verwendet, doch es hat sich bisher keine standardisierte Definition durchgesetzt. Darüber hinaus existiert eine Unmenge an verschiedenen Schreibweisen ([PGN06],[HTRS03]). Im Allgemeinen und unabhängig von der Schreibweise beschränkt sich die Definition des Begriffes *Online* auf die Verbindung eines Computers mit einem Netzwerk, anderen Computern oder dem Internet, über die er Daten senden und empfangen kann ([GB03]).

Im Zusammenhang mit der Simulation greift diese Definition zu kurz, da hier eine Verbindung zu anderen Systemen zu einem bestimmten Zweck erfolgt. Nach Hanisch et al. wird unter der *Online-Simulation* die Simulation eines real existierenden Systems verstanden, wobei das Simulationsmodell beim Start online (Anm.: also über die Verbindung zu einem Netzwerk) mit dem aktuellen Zustand des realen Systems initialisiert wird und die Simulationsergebnisse in hinreichend kurzer Zeit präsentiert werden ([HTRS03]). Diese Definition ist aus mehreren Gründen einerseits zu einschränkend und andererseits nicht spezifisch genug.

Bei der *Online-Simulation* geht es um die Versorgung eines Simulationsmodells mit Daten über das reale System. Dies geschieht über die Verbindung mit anderen Computern, über ein

Netzwerk oder das Internet. Die Repräsentation des aktuellen Systemzustandes ist dabei nicht zwangsläufig notwendig. So gehört auch die Simulation gespeicherter historischer Ereignisse oder die Simulation mit Durchschnittswerten zur Online-Simulation, wenn die Datenversorgung über eine solche Verbindung geschieht ([PGN06],[Dav98]).

Weiterhin ist es nicht notwendig, die Ergebnisse der Simulation in hinreichend kurzer Zeit zu repräsentieren. Die Versorgung des Simulationsmodells mit Daten ist unabhängig von der Repräsentation der Ergebnisse. Erst wenn die Online-Simulation zu einem bestimmten Zweck eingesetzt wird, ist dies in einen zeitlichen Zusammenhang zu betrachten. Soll bspw. die Produktionssteuerung eines realen Materialflusssystemes unterstützt werden, so müssen die Ergebnisse in einem bestimmten zeitlichen Rahmen zur Verfügung stehen.

Die Versorgung der Simulationsmodelle mit Daten dient zur Initialisierung des Modells mit dem aktuellen Systemzustand, zur Veränderung wichtiger Systemparameter oder zur Durchführung von Modellupdates vor einem Simulationslauf. Dies kann aber auch während eines Simulationslaufes geschehen (bspw. zu definierten Zeitpunkten). So findet etwa eine permanente Aktualisierung eines Simulationsmodells statt, dessen Zeitfortschritt in Echtzeit oder echtzeitproportional erfolgt.

Folgende allgemeine Definition der Online-Simulation berücksichtigt diese Sachverhalte:

„Unter der Online-Simulation versteht man die Simulation eines real existierenden Systems. Dabei kann das Simulationsmodell sowohl beim Start als auch während des Simulationslaufes online mit Daten über dieses System zur Initialisierung oder Aktualisierung versorgt werden.“

Im Zusammenhang mit der Online-Simulation werden im Folgenden noch verschiedene Begriffe näher betrachtet und von dem Themengebiet SEWS abgegrenzt.

3.2.3.1 Hardware-In-The-Loop- und Real-Time-Simulation

Bei *Hardware-In-The-Loop (HIL)*-Simulationen werden die Funktionalitäten eines Steuergerätes (Hardware) mit Hilfe eines Simulationsmodells vor dessen Einsatz in einem realen System überprüft ([KV00]). Dabei interagiert das Steuergerät mit einem Simulationsmodell, welches das zu steuernde System ersetzt. Diese Vorgehensweise wird auch als Emulation bezeichnet, wobei ein System durch ein anderes System funktional nachgebildet wird. Die Hardwarekomponenten und das Simulationsmodell bilden gemeinsam das zu entwickelnde reale System, wobei HIL-Simulationen je nach Problemstellung sowohl in Echtzeit bzw. echtzeitproportional als auch zeitlich gerafft oder verlangsamt ablaufen können. Der überwiegende Einsatz in der Automobilindustrie erfolgt zur Entwicklung und zum Test von Fertigungssteuerungssystemen oder bei mechatronischen Systemen (bspw. Assistenzsysteme wie *Steer-by-wire*, *Break-by-wire* oder automatische Einparkhilfen) ([SSSI05]).

Simulationsmodelle, deren Zeitfortschritt in Echtzeit oder echtzeitproportional erfolgt, werden auch als Real-Time-Simulationen bezeichnet. Die bereits angesprochenen HIL-Simulationen gehören ebenfalls zur Real-Time-Simulation, wenn der Zeitfortschritt in Echtzeit geschieht.

Darüber hinaus existieren aber auch Echtzeitsimulationen in Kombination mit einer Software, so dass es auch Echtzeitsimulationen gibt, die keine HIL-Simulationen sind.

3.2.3.2 Betreibersimulation

Der Begriff der *Betreibersimulation* findet keine einheitliche Verwendung und wird in der Literatur unzureichend diskutiert. Nach Wortmann werden unter der Betreibersimulation die Anlagensimulationen, die Fördertechniksimulation und die technische Anlaufsimulation im realen Betrieb zusammengefasst ([Wor02a]). Andere Autoren verstehen darunter ein Web-basiertes System zum „*Verbergen der Komplexität heutiger Simulationsumgebungen*“ ([Vor03]). Diese Ansätze beschreiben den Begriff der Betreibersimulation nicht vollständig bzw. sind in einem anderen Kontext zu sehen.

In der VDI-Richtlinie 3633 wird die Simulation in der Betriebsphase beschrieben ([VDI00]). Hierzu gehört der Einsatz der Simulation während des Betriebes eines realen Materialflusssystemes zur Beantwortung typischer Fragestellungen. Dies können Variantenuntersuchungen zur operativen Entscheidungsfindung oder die Überprüfung von Notfallstrategien oder Sofortmaßnahmen sein. Betreiber eines realen Materialflusssystemes sind mit der Steuerung bzw. dem Management des Systems beauftragt. Dabei werden vor allen Dingen die Aufgaben eines PPS-Systems unterstützt (vgl. Abschnitt 2.2.1.2). Verknüpft man diese Inhalte mit dem Begriff der Betreibersimulation, so fasst dieser die Unterstützung eines Betreibers eines realen Systems mit Hilfe der Simulation zusammen.

Hierzu zählen auch die so genannten Monitoringsysteme, die permanent den Zustand eines realen Materialflusssystemes erfassen und überwachen. Dabei wird ein Alarmmodul eingesetzt, welches Ausnahmestände und Fehlermeldungen im realen System registriert ([Fel00]). Hierzu werden zuvor mögliche Ausnahmefälle wie etwa Maschinenstillstände in einer Fehlerdatenbank definiert und verwaltet. Anschließend wird eine Maßnahmenbewertung in Kooperation mit einem Anwender durch Simulationsexperimente mit Hilfe eines Simulationsmodells durchgeführt.

3.2.3.3 Leitstandssimulation

Werden Simulationsmodelle mit Daten aus einem Leitstand versorgt, so handelt es sich um die so genannte *Leitstandssimulation*. Dabei geht es nicht um die genaue Abbildung des aktuellen Zustandes des realen Systems, sondern um den Einsatz der Simulation in Bereichen, in denen die analytischen Methoden eines Leitstandes nicht mehr ausreichen ([SPJS99]). Vor allen Dingen mittelfristige Kapazitäts- und Belegungsplanungen mit Hilfe eines Feinplanungssystems befinden sich im Fokus der Leitstandssimulation.

Als Feinplanungssystem werden *Scheduling-Systeme* oder die *Simulationsgestützte Ablaufplanung* (*SBS: Simulation-based Scheduling*) verwendet ([Wei06]). Scheduling-Systeme nutzen Algorithmen und Heuristiken zur Erzeugung zulässiger und optimaler Ablaufpläne. Dabei werden verschiedene Restriktionen wie Kapazitätsbeschränkungen und Vorgänger-Nachfolger-Beziehungen berücksichtigt. Die Schwäche von Scheduling-Systemen liegt in der Komplexität

der verwendeten Algorithmen bzw. Heuristiken, die von der Art und der Anzahl der Restriktionen abhängig ist ([NM02]).

Dagegen können bei der Simulationsgestützten Ablaufplanung wesentlich mehr Details und komplexe Ablaufregeln abgebildet werden. Dabei wird ein Simulationsmodell des realen Systems als eingebettete Applikation verwendet ([FR04]). Ein weiterer Vorteil liegt in der von Ressourcen selbständig durchgeführten Überwachung ihrer Kapazitätsgrenzen, wodurch von vornherein die Überlastung von Lagern oder Maschinen ausgeschlossen wird.

Im direkten Vergleich zu Scheduling-Systemen liefert das SBS keine optimalen Ablaufpläne. Oftmals wird nur ein zulässiger Ablaufplan gefunden, der allerdings eine verlässlichere Terminplanung ermöglicht. Dies liegt an dem stärkeren Realitätsbezug der in einem Simulationsmodell verwendeten Steuerparameter als dies bei Scheduling-Systemen der Fall ist.

3.2.3.4 Einordnung Simulationsbasierter Frühwarnsysteme

HIL-Simulationen und simulationsbasierte Leiststände können unter den Begriff der Online-Simulation eingeordnet werden, da sie ein Simulationsmodell mit einem real existierenden System verbinden und mit diesem Daten austauschen. Dagegen benötigen sowohl die Betreibersimulation als auch die Real-Time-Simulation nicht zwangsweise die permanente Verknüpfung mit einem realen System. Da HIL-Simulationen und Real-Time-Simulationen in Echtzeit ablaufen können, überschneiden sich diese beiden Themengebiete.

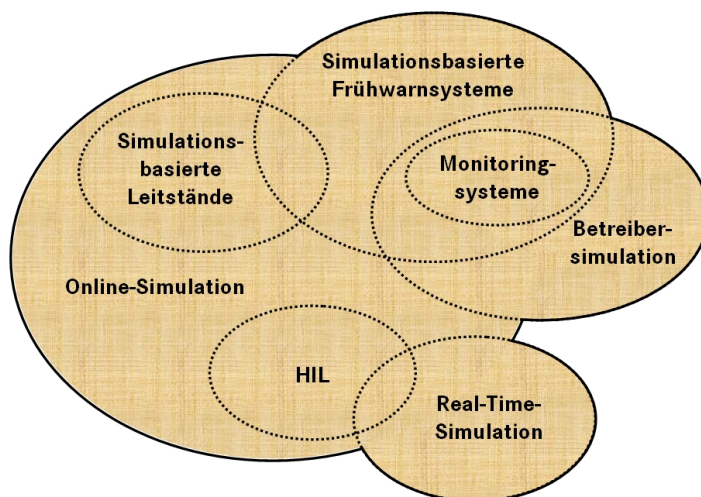


Abb. 3.1: Einordnung des Themengebietes SEWS

Von den HIL- und Real-Time-Simulationen können SEWS definitiv abgegrenzt werden. Zum einen besteht die Zielsetzung nicht in der Verknüpfung mit einer Hardware, um deren Funktionalitäten zu überprüfen. Weiterhin ist die Echtzeitsimulation für SEWS aufgrund der gewünschten Prognosefunktionalität nicht sinnvoll, da die Simulationsmodelle zukünftige Ereignisse prognostizieren sollen.

Dagegen decken SEWS die Funktionalitäten von Monitoringsystemen ab und haben deshalb gemeinsame Anteile mit der Betreibersimulation. Da ein SEWS den aktuellen Zustand des realen Systems auf Ausnahmefälle hin untersuchen und den Anwender durch generierte Handlungsalternativen unterstützen soll (vgl. Anforderung der Reaktivität in Abschnitt 3.2.2), sind die Funktionalitäten von Monitoringsystemen implizit enthalten. Ein SEWS soll allerdings nicht primär zur Steuerung eines realen Systems herangezogen werden und bspw. zur Belegungsplanung dienen. Wie simulationsbasierte Leitstände, benötigen auch SEWS aktuelle Daten aus Leistungsdatenbanken. Im Gegensatz dazu werden diese zur Initialisierung und Aktualisierung der Simulationsmodelle verwendet, um den aktuellen Zustand des realen Systems abzubilden.

SEWS besitzen also Bestandteile der Online-Simulation, gehören allerdings nicht vollständig zu dieser Gruppe. Dies liegt an den Anforderungen des reaktiven und proaktiven Verhaltens, die in der Definition der Online-Simulation nicht explizit enthalten sind. Abbildung 3.1 verdeutlicht diese Zusammenhänge nochmals.

3.3 Eine allgemeine Architektur Simulationsbasierter Frühwarnsysteme

In diesem Abschnitt wird eine allgemeine Architektur Simulationsbasierter Frühwarnsysteme zum Einsatz in der Produktion und Logistik vorgestellt. Es erfolgt eine Orientierung an den in Abschnitt 3.2.2 erläuterten Anforderungen an SEWS.

Dabei soll unter einer Architektur die Beschreibung der grundlegenden Elemente und der Struktur eines Softwaresystems verstanden werden. Balzert beschreibt sie als

„[...] eine strukturierte oder hierarchische Anordnung der Systemkomponenten, so- wie die Beschreibung ihrer Beziehungen.“ ([Bal01])

Bei der Entwicklung eines Systems stellt der Architekturentwurf die früheste Softwaredesign-Entscheidung dar und beeinflusst wichtige Eigenschaften wie die Modifizierbarkeit, die Wartbarkeit, die Sicherheit und die Performance des Softwaresystems. Die Entscheidung über deren Design ist eine der wichtigsten und kritischsten Momente bei der Softwareentwicklung, da die Softwarearchitektur im Nachhinein nur mit erheblichem Aufwand verändert werden kann ([Bal01]).

Im Folgenden wird zunächst auf die erforderlichen Hauptkomponenten eines SEWS eingegangen, wonach die Erläuterung des Aufbaus der Hauptkomponente Framework erfolgt.

3.3.1 Hauptkomponenten eines SEWS

Die Architektur eines SEWS besteht aus mehreren Hauptkomponenten, die sich aus den folgenden grundsätzlichen Voraussetzungen für einen sinnvollen Einsatz ergeben:

- Aussagekräftige Daten über den aktuellen Systemzustand des zu überwachenden Materialflusssystemes.

- Ein oder mehrere geeignete Simulationsmodelle, welche die Eigenschaften des Systems in ausreichender Weise beschreiben.
- Ein Anwender des Simulationsbasierten Frühwarnsystems.

Die erforderlichen Datenquellen stellen die erste Komponente dar und müssen den aktuellen Systemzustand, den geplanten Zustand und Historiedaten des realen Materialflusssystems beinhalten. Die Vorhersage künftiger Systemzustände des realen Systems wird mit Hilfe der zweiten Komponente, den Simulationsmodellen erreicht. Der dritten Komponente in Form eines Anwenders des SEWS werden die Prognoseergebnisse und vorgeschlagenen Handlungsalternativen zur Verfügung gestellt, der dann geeignete Gegenmaßnahmen in dem realen System umsetzt. Diese Komponenten müssen miteinander in geeigneter Weise interagieren und kommunizieren, weshalb eine Framework als vierte Komponente notwendig ist.

Das Framework übernimmt wichtige Kernfunktionalitäten des SEWS. Dabei handelt es sich um die Koordination der Komponenten untereinander, der Steuerung der Simulationsmodelle, dem Datenhandling, der Datenbeschaffung und der Kommunikation mit dem Anwender. Die Hauptkomponenten eines SEWS werden nochmals in Abbildung 3.2 dargestellt.

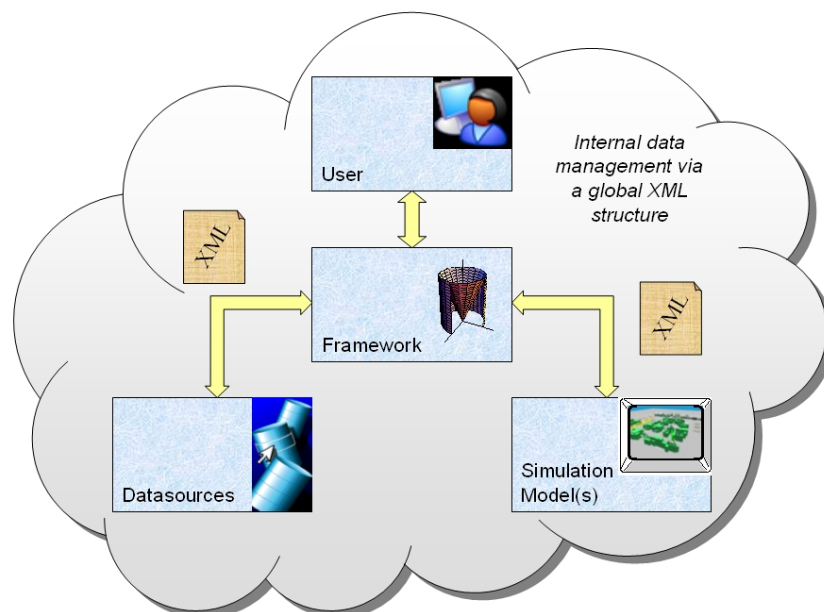


Abb. 3.2: Die Hauptkomponenten eines Simulationsbasierten Frühwarnsystems

Um den Anforderungen Erweiterbarkeit, Wiederverwendbarkeit und Standardisierung Rechnung zu tragen, wird zum SEWS-internen Datenmanagement XML verwendet. Die *eXtensible Markup Language (XML)* ist eine vom *World Wide Web Consortium (W3C)* definierte standardisierte Auszeichnungssprache zur Erstellung maschinen- und menschenlesbarer Dokumente in Form einer Baumstruktur ([W3C06]). XML ist eine Metasprache, mit der andere Sprachen für eine spezifische Anwendung beschrieben werden können. Als Weiterentwicklung der *Standard Generalized Markup Language (SGML)* spielt sie eine bedeutende Rolle als

universelles und plattformunabhängiges Datenaustauschformat zwischen verschiedenen Systemen und Softwarekomponenten ([Möh99]). Der Grundgedanke von XML liegt in der Trennung von Daten und ihrer Repräsentation. Dadurch wird der einfache Austausch der einzelnen Komponenten eines SEWS und die Erweiterung oder Veränderung von Komponentenfunktionalitäten unterstützt ([McL02],[Sku04],[Bat03]). Ein XML-Dokument muss wohlgeformt und valide sein. Valide XML-Dokumente sind zu einem XML-Schema konform, wenn dieses zu den in diesem XML-Schema definierten Wertebereichen und Positionierung von Daten konform ist.

Die zu verwendenden Datenquellen für die Bereiche Produktion und Logistik sind überwiegend heterogen und abhängig vom entsprechenden realen System. Dabei kann es sich um relationale Datenbanken handeln, die Produktionssteuerungssysteme oder Produktionsleitstände versorgen. Aber auch Produktionsplanungssysteme wie *Enterprise Resource Planning-Systeme (ERP-Systeme)* oder andere Planungs- und Dokumentationsdatenbanken enthalten relevante Daten für ein SEWS. Zusätzlich werden vielfach Instandhaltungsdaten und Daten aus Controllingssystemen benötigt. Je nach Anwendungsfall sind die für den Einsatz eines SEWS relevanten Datenquellen in Abhängigkeit verschiedener Kriterien zu identifizieren. Bestimmte Szenarien machen es erforderlich, direkt Daten von Förderbändern oder Maschinen über *NC-Steuerungen, speicherprogrammierbare Steuerungen (SPS)* oder *OPC-Servern* zu verwenden. Hierzu existieren bereits Ansätze, um Realdaten von Produktionseinheiten direkt mit Simulationsmodellen zu koppeln ([Fel00]) und simulationsrelevante Daten in XML zu repräsentieren ([RNJ03]).

Die Anforderung der Unabhängigkeit gilt auch im Zusammenhang mit spezifischen Simulatoren oder Simulationsmodellen. Diese müssen ohne Beeinflussung der anderen Hauptkomponenten austauschbar sein. Geeignete Simulatoren sind hinsichtlich verschiedener Kriterien wie etwa deren Erweiterbarkeit, sowie Kommunikations- und Leistungsfähigkeit auszuwählen.

Eine ausreichende Performance von SEWS ist eine weitere Anforderung, die sehr stark an die Geschwindigkeit von Simulationsläufen eines Simulators gekoppelt ist. Auf spezifische Anforderungen an Simulatoren für den geeigneten Einsatz in einem SEWS wird im nachfolgenden Kapitel eingegangen. Im Folgenden werden die erforderlichen Komponenten des Frameworks eines SEWS hergeleitet und erläutert.

3.3.2 Der Aufbau des Frameworks

Die Hauptkomponente Framework muss vielfältige Funktionen übernehmen, die sich zum einen aus den Anforderungen an SEWS ableiten und zum anderen aus der Integration der einzelnen Hauptkomponenten zu einer Gesamtarchitektur ergeben. Die Forderungen der Adaptionsfähigkeit, Skalierbarkeit, Unabhängigkeit, Erweiterbarkeit und Standardisierung führen zur einer notwendigen Gliederung des Frameworks in einzelne Komponenten, die spezifische Funktionen oder Funktionsgruppen beinhalten.

Da das Framework Schnittstellen zu den anderen Hauptkomponenten benötigt, führt dies zwangsläufig zu der Notwendigkeit von weiteren entsprechenden Komponenten im Frame-

work. Bei diesen Framework-Komponenten handelt es sich deshalb um einen *Data Listener*, eine *Simulation Control*-Komponente und ein *User-Interface*.

Data Listener

Die Komponente Data Listener ermöglicht die Erfassung relevanter Daten aus SEWS-externen Systemen und stellt diese in einem geeigneten Austauschformat zur Verfügung. Der Data Listener greift zur Erfassung der Daten auf die verfügbaren Datenquellen zu definierten Zeitpunkten, innerhalb definierter Zeitabstände oder in Abhängigkeit konkreter Ereignisse zu. Hierzu muss dem Data Listener bekannt sein, welche Datenquellen verwendet und wie aus ihnen Daten gewonnen werden können. Diese Informationen werden über ein User-Interface konfiguriert und zur Verfügung gestellt.

Simulation Control

Die Ansteuerung von Simulationsmodellen zur Erfüllung der Anforderungen Proaktivität und Reaktivität werden von einer weiteren Komponente, der Simulation Control, übernommen. Hierzu müssen Daten zwischen dem Framework und den Simulationsmodellen bidirektional ausgetauscht werden können. So sind zum einen dem Simulationsmodell simulationsrelevante Daten zu übergeben und zum anderen muss das Framework in der Lage sein, die Ergebnisse der Simulationsläufe zu empfangen. Darüber hinaus muss diese Komponente die Simulationsläufe steuern und nach Empfang der Ergebnisse diese aufbereiten und in geeigneter Form zur Verfügung stellen. Zur Ansteuerung von Simulatoren und Simulationsmodellen existieren bereits erfolgreiche Ansätze ([RJ03],[Fel00]).

User Interface

Eine Benutzerschnittstelle (User-Interface) ermöglicht die Kommunikation zwischen dem Anwender und einem SEWS. Darüber kann die Konfiguration und Steuerung des Frameworks, die Visualisierung und Verwaltung von Daten, die Adaption der einzelnen Module an spezifische Problemstellungen bzw. die vollständige Parametrierung des SEWS erfolgen. Sie stellt das Bindeglied zum menschlichen Faktor eines Frühwarnsystems dar, der entsprechend der in Abschnitt 3.1 vorgestellten generellen Komponenten eines Frühwarnsystems über die endgültige Einschätzung der Auswirkungen künftiger Ausnahmesituation und die Umsetzung von Gegenmaßnahmen im realen System entscheidet.

Das Framework eines SEWS benötigt noch weitere Komponenten, die aufgrund der Anforderungen und notwendigen Funktionalitäten eines SEWS erforderlich sind. Diese werden im Folgenden als *Exception Interpreter*, *Knowledge Base*, *Data Management* und *Generator of Alternatives* bezeichnet.

Exception Interpreter

Ein SEWS weist sowohl reaktives als auch proaktives Verhalten auf (vgl. Abschnitt 3.2.2). Es soll also bei gegenwärtigen, aber auch künftigen Ausnahmesituationen warnen. Hierzu ist es erforderlich, dass solche Ausnahmefälle erkannt und interpretiert werden können. Diese Aufgabe übernimmt die Komponente Exception Interpreter. Hierzu müssen logische Regeln zur

Verfügung stehen, die diese Interpretation und Detektion auf Basis von Daten über den aktuellen Systemzustand oder von Simulationsergebnissen zur Prognose künftiger Systemzustände ermöglichen.

Generator of Alternatives

Die Generierung von Handlungsalternativen beim Auftreten von Ausnahmesituationen und die erforderliche Evaluierung übernimmt die Komponente Generator of Alternatives. Unter Berücksichtigung der Performance sind bereits im Vorfeld ungeeignete Handlungsalternativen auszuschließen. Werden eine oder mehrere sinnvolle Handlungsalternativen vom System ermittelt und mit Hilfe der Simulation auf ihre Eignung hin überprüft, so wird der Anwender davon unterrichtet. Dieser kann dann entscheiden, ob er eine oder mehrere dieser Alternativen auswählt oder nicht.

Knowledge Base

Eine so genannte Knowledge Base ist eine Voraussetzung für den effizienten Einsatz eines SEWS. In einer solchen Wissensbasis werden definierte logische Regeln zur Interpretation von Daten hinsichtlich eventueller Ausnahmefälle und zur Generierung von Handlungsalternativen aufbereitet und verwaltet. Diese stehen dann den Komponenten Exception Interpreter und Generator of Alternatives zur Verfügung. Dabei ist es auch erforderlich, benutzerdefinierte Ausnahmesituationen wie bspw. zur Instandhaltung geplante Maschinenausfälle zu verarbeiten (User Exception, vgl. Abschnitt 3.2.2).

Die Regeln werden sowohl vom Benutzer direkt als auch durch das System selbst definiert. Zur Generierung sinnvoller Handlungsalternativen ist es aus Performancegründen wichtig, dass diese Wissensbasis die Reduzierung der Anzahl möglicher Handlungsalternativen unterstützt. Je geeigneter Handlungsalternativen erzeugt werden, desto weniger Rechnerressourcen werden in Anspruch genommen und desto schneller ist das Antwortzeitverhalten eines SEWS.

Data Management

Die für ein SEWS notwendigen Daten müssen in geeigneter Weise verwaltet werden. Die hierfür notwendigen Funktionalitäten werden durch die Komponente Data Management bereitgestellt. Die Funktionalitäten gehen über die simple Ablaufverfolgung des Frameworks hinaus, da sowohl tatsächliche und prognostizierte Ausnahmefälle als auch Handlungsalternativen archiviert werden müssen. Auch die Speicherung der definierten logischen Regeln wird durch diese Komponente übernommen.

Die hier beschriebenen Komponenten müssen in vielfältiger Weise miteinander interagieren und Informationen austauschen (vgl. Abb. 3.3), um die an ein SEWS gestellten Anforderungen zu erfüllen. Hierzu ist besonders eine verteilte Struktur geeignet, welche im folgenden Abschnitt näher beschrieben wird. Anschließend ist zu klären, wie die einzelnen Komponenten miteinander interagieren bzw. wie der Workflow zu gestalten ist.

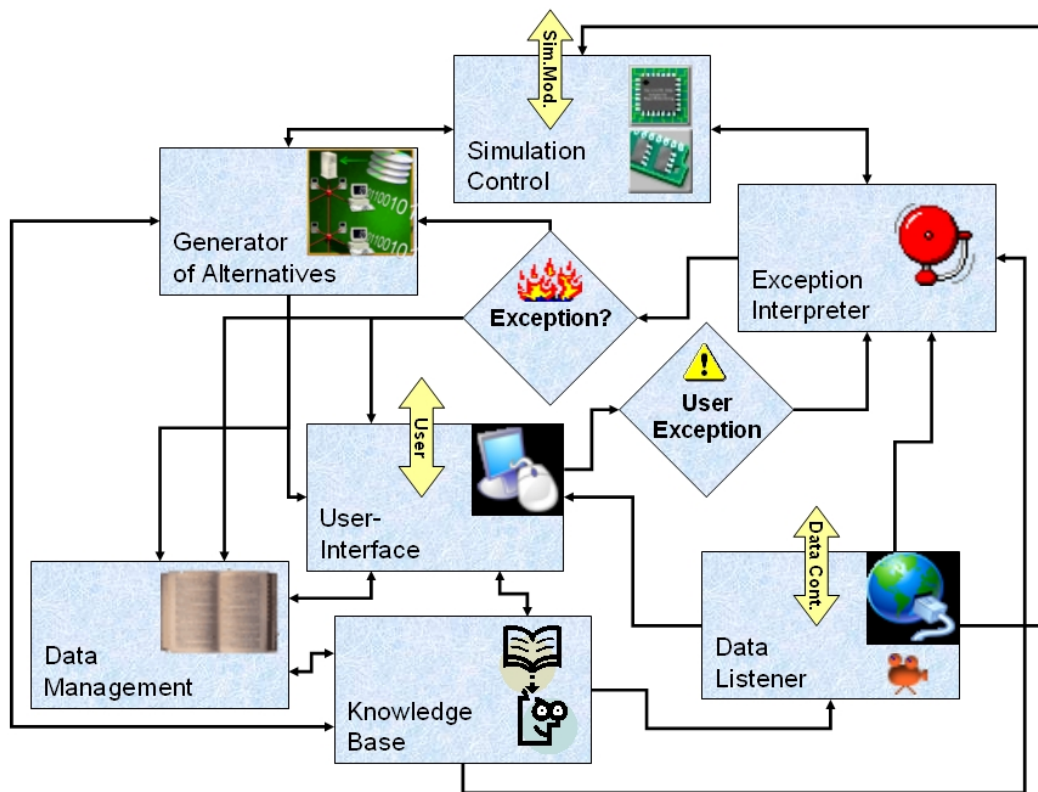


Abb. 3.3: Komponenten des Frameworks

3.4 Verteilte Systeme und Kommunikationsprinzipien

Die Komponenten eines SEWS haben unterschiedliche Funktionen und Aufgaben, die eine Kommunikation sowohl zwischen den Komponenten als auch komponentenintern notwendig machen. Aufgrund der Anforderung einer hohen Performance, muss ein effizientes Kommunikationsprinzip verwendet werden. Zur Prognose und zur Bewertung generierter Handlungsalternativen werden Simulationsläufe durchgeführt, die erhebliche Rechnerressourcen in Anspruch nehmen. Aus diesem Grund ist es sinnvoll, die Aufgaben und Funktionen der Komponenten mit Hilfe einer verteilten Struktur zur Verfügung zu stellen, um so die Rechenlast auf ein Netzwerk verteilen zu können. Auch die Anforderungen Erweiterbarkeit, Skalierbarkeit bzw. Adaptionfähigkeit, Standardisierung bzw. Wiederverwendbarkeit und Unabhängigkeit werden durch die Anwendung einer verteilten Struktur unterstützt.

Es existieren mehrere technische Möglichkeiten, verteilte Systeme zu realisieren. Im Allgemeinen ist ein verteiltes System ein Zusammenschluss unabhängiger Computer, welches sich dem Benutzer als ein einziges System präsentiert. Zu verteilten Systemen gehören beispielsweise *Client-Server-Systeme*, *Verteilte Anwendungen* und *Verteilte Betriebssysteme* ([TvS03]).

Vielfältige Gründe sprechen für den Einsatz verteilter Systeme. Ein Grund liegt in einer gewünschten echten Nebenläufigkeit von Prozessen zur Beschleunigung von Rechenzeiten. Ein weiterer wichtiger Grund ist die Skalierbarkeit durch hinzufügen und entfernen von Kompo-

nenten. Dadurch kann die Leistungsfähigkeit des verteilten Systems erhöht bzw. auf veränderte Rahmenbedingungen angepasst werden. Die Bereitstellung von entfernten Ressourcen ist ein weiteres Einsatzgebiet. Indem Funktionalitäten redundant von mehreren Rechnern angeboten werden, verringert sich die Fehleranfälligkeit eines verteilten Systems. Fällt ein Rechner in einem Netzwerk aus, so kann ein anderer dessen Funktionalitäten übernehmen.

Der folgende Abschnitt stellt die gängigen Technologien RPC, RMI, CORBA, DCOM und Web Services zur Vernetzung von Funktionalitäten und zum Aufbau eines verteilten Systems in einem Rechnernetzwerk vor. Dabei wird sowohl auf deren Funktionsweise, Vor- und Nachteile eingegangen als auch deren Eignung zum Einsatz in einem SEWS diskutiert.

3.4.1 RPC und RMI

Eine von *Sun Microsystems* entwickelte Technologie zur Unterstützung für verteilte Systeme ist der so genannte *Remote Procedure Call (RPC)*, der einen Funktionsaufruf auf entfernten Rechnern über ein Netzwerk ermöglicht. RPC basiert auf einem *Client-Server-Modell* zur gemeinsamen Nutzung von Programmfunktionen über Rechengrenzen hinweg. Ein Client-Server-Modell ist eine Netzwerkstruktur, bei dem Funktionalitäten von einem zentralen Server angeboten und von Clients in Anspruch genommen werden ([TvS03]). Die so genannte *Remote Method Invocation (RMI)* ist die Implementierung des RPC in der Programmiersprache *JAVA* ([MvNV⁺99]).

Mit Hilfe von RPC und RMI können rechnerübergreifende Methodenaufrufe sehr bequem durchgeführt werden. Der Aufwand für den Programmierer reduziert sich auf die Einbindung der entsprechenden Programmpakete, ohne sich bspw. mit einer Socketprogrammierung befassen zu müssen. Für RPC existiert keine Norm, sondern lediglich Standards, die durch Unternehmen wie *Sun* oder *Microsoft* festgelegt worden sind. Aus diesem Grund ist keine Plattformunabhängigkeit bezüglich unterschiedlicher Betriebssysteme gegeben. Weiterhin ist das RPC-Design speziell auf verschiedenen UNIX-Betriebssystemen sehr an der Programmiersprache *C/C++* orientiert, wodurch RPCs nicht uneingeschränkt über Sprachgrenzen hinweg agieren können ([Gun95]). Deshalb sind diese Technologien für den Einsatz in einem SEWS ungeeignet. Eine Weiterentwicklung für die verteilte Programmierung stellt CORBA dar, welches im Folgenden näher erläutert wird.

3.4.2 CORBA und DCOM

Die *Common Object Request Broker Architecture (CORBA)* ist eine von der *Object Management Group (OMG)* entwickelte objektorientierte Middleware zur einfachen Entwicklung verteilter Anwendungen in heterogenen Umgebungen ([NRS04]). Unter einer Middleware versteht man eine anwendungsunabhängige Technologie, die Dienstleistungen zur Vermittlung zwischen Anwendungen unter Verbergung der damit verbundenen Komplexität anbietet ([Kel02]). Hierfür definiert CORBA plattformübergreifende Protokolle und Dienste, wodurch die Unabhängigkeit von einer bestimmten Programmiersprache erreicht wird.

Im Zuge der objektorientierten Programmierung entstand CORBA aus dem Wunsch nach einem RPC-ähnlichen Mechanismus auf Objektebene heraus, der die Nachteile des RPC vermeiden sollte. Ähnliche Funktionalitäten wie CORBA bietet das *Distributed Component Object Model (DCOM)*, welches ein von *Microsoft* definiertes Protokoll zur Unterstützung der Kommunikation von Programmkomponenten über ein Netzwerk darstellt. Im Gegensatz zu CORBA ist DCOM allerdings nicht plattformunabhängig, sondern kommt auf Rechnern zum Einsatz, die mit einem Betriebssystem des Unternehmens *Microsoft* arbeiten.

Die Nachteile von CORBA liegen sowohl in einer schwierigeren Implementierung als auch in der mangelhaften Unterstützung architektureller Problemstellungen bei der Entwicklung großer verteilter Objektsysteme ([BR97]). CORBA befreit den Programmierer von vielen so genannten *Low-Level-Details* und bietet viele Einzellösungen an, allerdings keine Architekturversion. Weiterhin unterstützten CORBA und DCOM die Kommunikation über Firewalls hinweg nur unzureichend. Aus diesem Grund ist auch CORBA und DCOM für SEWS ungeeignet. Im folgenden Abschnitt werden Web Services als letzte Technologie zum Aufbau verteilter Systeme im Rahmen dieser Arbeit vorgestellt.

3.4.3 Web Services

Ein Web Service ist eine über einen *Uniform Resource Identifier (URI)* eindeutig identifizierbare Softwareanwendung zur Unterstützung der Interoperabilität verschiedener Softwareapplikationen in einem Netzwerk unterschiedlicher Rechner und Plattformen ([GSB⁺02]). Sie ermöglichen die einfache Kombination verschiedener Softwarekomponenten und Dienste zu einem integrierten Service oder zur Wiederverwendung in einer weiteren Infrastruktur. Dabei orientieren sie sich an den Grundgedanken einer *Service-Oriented Architecture (SOA)*, die aus der Identifikation von Diensten, der Trennung von Zuständigkeiten nach fachlichen Gesichtspunkten und der Datenkapselung bestehen. Die SOA setzt zunächst ein Managementkonzept voraus, welches eine an den Geschäftsprozessen ausgerichtete IT-Infrastruktur zum Ziel hat. Anschließend wird ein Systemarchitekturkonzept benötigt, welches fachliche Dienste und Funktionalitäten in Form von Webdiensten zur Verfügung stellt ([Erl04]).

Web Services besitzen eine Schnittstelle, die mit Hilfe eines auf XML basierenden, maschinenlesbaren Formats, der *Web Services Description Language (WSDL)* beschrieben wird. Die Veröffentlichungen eines Web Service findet mit Hilfe eines so genannten *Service Broker* statt, der ein Verzeichnis von Diensten und deren Beschreibung anbietet. Hierzu dient wiederum der Standard *UDDI (Universal Description, Discovery and Integration)* zur Registrierung von Web Services bei einem Verzeichnisdienst. Dadurch kann ein Servicekonsument (*Service Requester*) einen für eine spezifische Aufgabe geeigneten Dienst finden und erhält Informationen über den entsprechenden Serviceanbieter (*Service Provider*). Hierzu verwenden Systeme, die mit Web Services interagieren, das *Simple Object Access Protocol (SOAP)*, welches ein Protokoll zum Austausch von Nachrichten über ein Computernetzwerk darstellt ([Liv02]). Dabei verwendet SOAP XML als Standard zur Repräsentation der Daten und Internet-Protokolle der Transport- und Anwendungsschicht zur Übertragung dieser Nachrichten ([STK02]).

Die Nachteile von Web Services liegen in möglichen Performanceschwächen, da durch die Verwendung von XML und SOAP ein erheblicher Overhead entsteht und der Verwaltungsaufwand bei stark verteilten Systemen zunimmt. Dieser Nachteil kann durch eine effiziente Wahl von XML-Parsern, dem Kommunikationsprinzip und der Verteilung der Komponenten in einem Netzwerk ausgeglichen werden. Weiterhin sind besonders Sicherheitsaspekte zu berücksichtigen, da in einem Netzwerk verfügbar gemachte Dienste durch Authentifizierung und Verschlüsselung vor Angriffen aus dem Internet gesichert werden müssen ([Knu03]). Ein weiterer Nachteil ist die Notwendigkeit eines größeren Know-Hows bei der Programmierung, Anwendung und Einbindung von Web Services zur Entwicklung verteilter Systeme, da spezielle Programmbibliotheken eingebunden und Schnittstellen genau definiert werden müssen. Allerdings ist dies bei vergleichbaren Technologien ebenfalls der Fall.

Den Nachteilen stehen sehr große Vorteile entgegen. So entsteht durch die Verwendung offener, weit verbreiteter und bestehender Internet-Standards eine flexible Architektur. Diese ist unabhängig von verwendeten Plattformen, Programmiersprachen und Protokollen und unterstützt somit die Interoperabilität von Softwareanwendungen und -applikationen. Dadurch können in einfacher Art und Weise komplexe verteilte Systeme aufgebaut werden, deren architekturelle Probleme durch die Anwendung von Web Services gelöst werden. Weiterhin fallen durch die Verwendung dieser offenen Standards keinerlei Lizenzkosten an. Darüber hinaus verhindert die Verwendung des HTTP-Protokolls zur Datenübertragung im Vergleich zu RPC, RMI, CORBA und DCOM Probleme mit eingesetzten Firewalls in einem Netzwerk oder Internet.

3.4.4 Eingesetztes Kommunikationsprinzip

Aufgrund ihrer Vorteile bieten sich Web Services zur Anwendung bei der Entwicklung und dem Aufbau von SEWS an. Rechenintensive Simulationsläufe und die Datenbeschaffung werden auf verschiedenen Rechnern in einem Netzwerk durchgeführt. Die unterschiedlichen Komponenten des Frameworks können als Web Service angeboten und anschließend zur *Gesamtanwendung* SEWS zusammengefasst werden. Die mögliche und notwendige Kapselung und Verteilung der einzelnen Funktionalitäten eines SEWS in einem Netzwerk ist stark von der vorhandenen Unternehmensinfrastruktur abhängig. Die verwendeten Rechner müssen mit einem Webserver ausgestattet werden, auf dem der oder die Web Services installiert werden. Dabei ist ein Webserver im engeren Sinne ein Server-Dienst, der Informationen über das *HyperText Transfer Protocol (HTTP)* zur Verfügung stellt ([Knu03]). Ein Webserver dient aber auch als Container für Webanwendungen, Servlets und Web Services.

Um die Anforderung der Performance an ein SEWS zu erfüllen, muss der zunehmende Kommunikationsaufwand bei stärkerer Verteilung der Funktionalitäten über ein Netzwerk gegen die Verteilung der Rechenlast auf mehrere Rechner abgewogen werden. Für SEWS macht es Sinn, zentral zu verwaltende Komponenten des Frameworks wie der Generator of Alternatives, das Data Management und das User-Interface auf einem so genannten *Master Web Server* anzubieten. Auf diesem Server werden die verteilten Web Services eingebunden und zusammen mit den zentralen Komponenten als integrierter Dienst eines SEWS zur Verfügung gestellt.

Abbildung 3.4 zeigt die verteilte Organisation der Komponenten eines SEWS. Hierbei ist der Master Web Server als *Main Controller* gekennzeichnet, der die SEWS-Komponenten Generator of Alternatives, Data Management und Knowledge Base beinhaltet. Die Komponente Data Listener wird auf mehrere Web Services verteilt, die eine parallele Datensammlung von verschiedenen Datenquellen ermöglichen. Eine Einschränkung bei dieser Parallelisierung ergibt sich durch eventuelle Schlüsselinformationen, die zunächst aus einer Datenquelle ermittelt werden müssen, um relevante Daten aus einer anderen Datenquelle erhalten zu können.

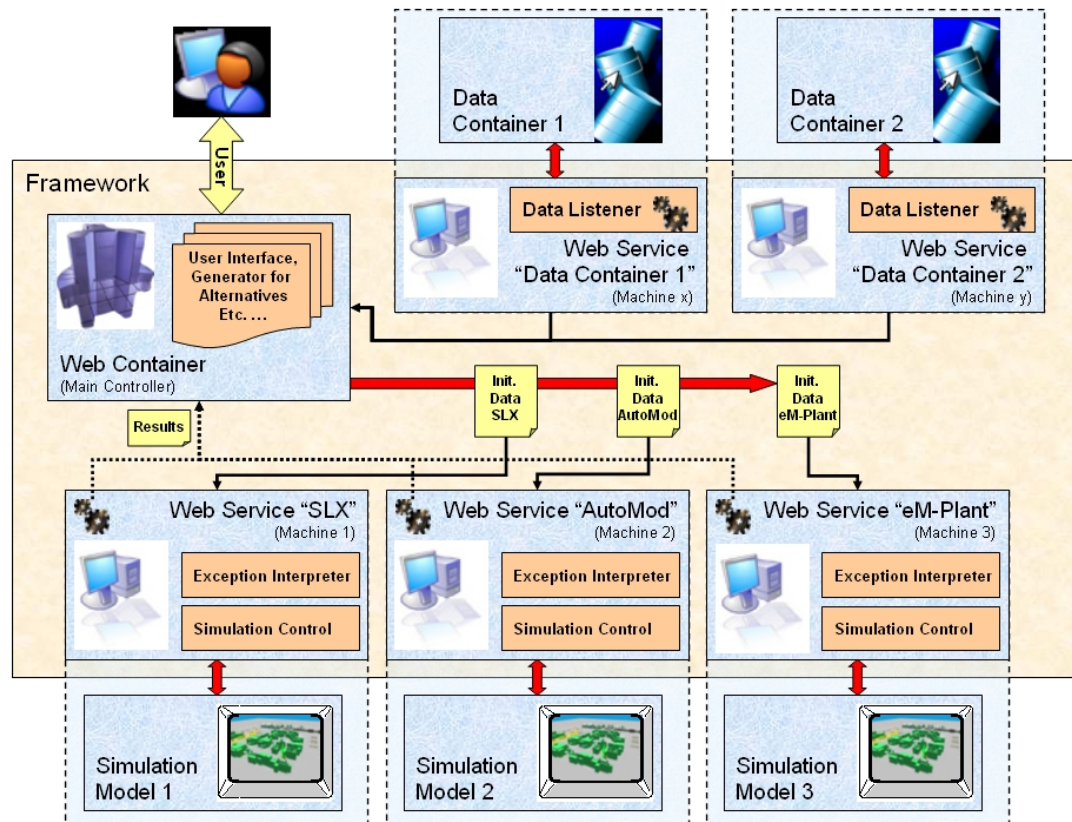


Abb. 3.4: Verteilte Organisation der Komponenten eines SEWS

Aufgrund der Anforderung der Unabhängigkeit von SEWS (vgl. Abschnitt 3.2.2) sollen verschiedenste Simulationsmodelle bzw. Simulatoren in einfacher Weise eingebunden werden können. Deshalb und aus Performancegründen wird die Prognosefunktionalität mit Hilfe der Simulation ebenfalls als Web Service auf verschiedenen Rechnern verteilt zur Verfügung gestellt. Hierzu werden die Funktionen der SEWS-Komponente Simulation Control aus dem Main Controller ausgelagert und von einem simulatorspezifischen Web Service (bspw. *Web Service SLX* oder *Web Service AutoMod*) angeboten. In Abbildung 3.4 ist die Komponente Exception Interpreter als simulatorspezifische Funktion eines Web Service dargestellt. Dies ist sinnvoll, da während des Simulationslaufes zu bestimmten Kontrollzeitpunkten oder kontinuierlich eine Untersuchung des Modellzustandes hinsichtlich möglicher Ausnahmefälle durchzuführen ist. Je länger der Prognosezeitraum und je geringer die Distanz zwischen den simulationsinternen Kontrollzeitpunkten, desto größer würde der Kommunikationsaufwand zwischen dem Main Controller und dem Web Service werden.

Im Verlauf der Entwicklung von SEWS zeigte sich weiterhin, dass dieser Kommunikationsaufwand auch Web Service-intern (Anm.: also die Kommunikation zwischen Simulationsmodell und dem Exception Interpreter über die Komponente Simulation Control) zu gross wurde und zu Performanceverlusten führte. Aus diesem Grund muss die Komponente Exception Interpreter als Funktion in jedem Simulationsmodell zur Verfügung stehen. Wie diese Herausforderung bei der Entwicklung von SEWS bewältigt werden kann, wird im nachfolgenden Kapitel näher erläutert.

Die von einem Web Service angebotenen Funktionalitäten können erst durch dessen Aufruf über seine eindeutige URI in Anspruch genommen werden. Ein Web Service handelt also passiv, wodurch sich die Fragestellung nach der zeitlichen Steuerung der Funktionalitäten der Komponenten eines SEWS ergibt. Eine Erläuterung, wie der Workflow in einem SEWS gesteuert werden kann, wenn die einzelnen Komponenten mit Hilfe eines Web Service gekapselt und angeboten werden, erfolgt im folgenden Abschnitt.

3.5 Steuerung des Workflows

Es ist offensichtlich, dass innerhalb eines SEWS eine vielfältige und komplexe Kommunikation zwischen den Komponenten erfolgen muss. Die in Abschnitt 3.4.3 dargelegten Vorteile von Web Services zeigen, dass diese besonders zur Entwicklung komplexer verteilter Systeme geeignet sind. Das überwiegende Einsatzgebiet liegt in der Unterstützung von Geschäftsprozessen, worin Web Services einzelne Aktivitäten repräsentieren. Bei den hiermit verbundenen Ansätzen handelt es sich um die so genannte *Enterprise Application Integration (EAI)* oder der Bereich *Business-To-Business (B2B)*.

Die EAI ist ein Konzept zur unternehmensweiten Integration der Geschäftsfunktionen entlang der Wertschöpfungskette ([Kel02]). Dabei entsprechen die Geschäftsfunktionen verschiedenen Applikationen, die auf unterschiedlichen Plattformen bzw. Rechnern in einem Unternehmensnetzwerk verteilt sind. EAI findet vor allem im Bereich des E-Business in Form von unternehmensinternen Portalen Anwendung ([Kel02]). Verstärkt in Großunternehmen werden dadurch komplexe IT-Landschaften miteinander mit dem Ziel verknüpft, dem Kunden ein konsistentes Bild des Unternehmens und seiner Interaktionen zu bieten. Hier ersetzen Web Services die klassischen Middleware-Produkte (vgl. Abschnitt 3.4).

Neben dem EAI liegt ein weiteres Haupteinsatzgebiet im Bereich des B2B, womit die Kommunikationsbeziehungen zwischen mindestens zwei Unternehmen beschrieben wird. B2B ist der älteste und größte Bereich des E-Business und dient speziell zur Unterstützung von interorganisationellen Geschäftsprozessen mit dem Schwerpunkt des elektronischen Ein- und Verkaufs von Artikeln.

Im folgenden Abschnitt werden zunächst die notwendige Interaktion der Komponenten und ein Ablaufplan des notwendigen Workflows eines SEWS vorgestellt.

3.5.1 Interaktion der Komponenten eines SEWS

Die in Abschnitt 3.3.2 dargestellten Komponenten des Frameworks stellen unterschiedliche Funktionalitäten zur Verfügung, die in ihrer Gesamtheit den Anforderungen an ein SEWS genügen müssen. Es stellt sich nun die Frage, wie und wann diese Komponenten miteinander interagieren. Hierzu wird die in Abbildung 3.3 bereits angedeuteten Kommunikation näher erläutert.

Grundlage für ein SEWS sind die Daten über das reale System, die über den Data Listener aus unterschiedlichen Datenquellen gewonnen und den anderen Frameworkkomponenten zur Verfügung gestellt werden. Zum einen ist die Komponente Simulation Control mit diesen Daten zur Prognose künftiger Systemzustände zu versorgen. Zum anderen sind die Daten über den aktuellen und den prognostizierten Systemzustand dem Anwender des SEWS über das User Interface durch eine geeignete Visualisierung zur Verfügung zu stellen.

Die Konfiguration des Data Listeners und die Definition der zu verwendenden Datenquellen wird in der Komponente Knowledge Base hinterlegt. Hier finden sich auch die logischen Regeln, die der Exception Interpreter zur Analyse der Daten über den aktuellen Systemzustand und der mit Hilfe der Simulation prognostizierten künftigen Systemzustände benötigt. Diese Regeln und die Informationen zur Konfiguration des Data Listeners erhält die Knowledge Base über die Benutzerschnittstelle.

Tritt ein Ausnahmefall auf, so wird zeitgleich der Anwender über die Benutzerschnittstelle informiert und die Komponente Generator of Alternatives aktiviert. Diese kommuniziert mit der Knowledge Base, die Regeln zur Generierung von Handlungsalternativen in Abhängigkeit des Ausnahmefalls beinhaltet. Mit Hilfe dieser Regeln werden Handlungsalternativen erzeugt und Simulationsläufe gestartet. Anhand der Simulationsergebnisse erfolgt eine Bewertung dieser Handlungsalternativen hinsichtlich ihrer Wirksamkeit zur Vermeidung von Ausnahmefällen.

Die eigentliche Datenverwaltung wird mit Hilfe des Data Management erreicht. Hier sind sowohl historische und aktuelle Daten über das System, über generierte Handlungsalternativen und prognostizierte Systemzustände als auch über die Inhalte der Knowledge Base hinterlegt. Die Inhalte dieser Komponente werden mit Hilfe der Benutzerschnittstelle visualisiert, konfiguriert und verwaltet.

Abbildung 3.5 zeigt einen Ablaufplan des Workflows eines SEWS. Dabei ist die Datenbeschaffung aus unterschiedlichen Datenquellen parallelisierbar. Dies gilt auch für die Bewertung der unterschiedlichen Handlungsalternativen. An dieser Stelle wird deutlich, dass der Generator of Alternatives möglichst sinnvolle Alternativen erzeugen muss. Ungeeignete Alternativen liefern keine ausnahmsfreie Prognose eines künftigen Systemzustandes, wodurch weitere Handlungsalternativen zu generieren sind. Es ist leicht nachzuvollziehen, dass sehr viele ungeeignete Handlungsalternativen die Performance eines SEWS negativ beeinträchtigen. Dies ist auch durch eine Parallelisierung der Bewertung möglicher Handlungsalternativen nur bedingt zu beeinflussen.

Der Ablaufplan stellt die zeitliche Abfolge der Aktivitäten bei der einmaligen Prognose und Generierung von Handlungsalternativen auf Basis eines zu einem bestimmten Zeitpunkt herr-

der Entwicklungsphase von SEWS zum Einsatz kam. Anschließend wird die *Business Process Execution Language (BPEL)* als Vertreter einer möglichen Beschreibungssprache vorgestellt. Zum Abschluss soll auf die Thematik der zyklischen Durchführung der Abläufe eingegangen werden.

3.5.2.1 Spezifische Anwendungslösung

Bei der Entwicklung und dem Design von SEWS im Rahmen dieser Arbeit wird die Programmiersprache *Java* verwendet. Hierbei handelt es sich um eine objektorientierte Programmiersprache der Firma *Sun Microsystems*. *Java* ist ohne weitere Anpassungen auf verschiedenen Rechnern und Betriebssystemen lauffähig, auf denen eine *Java Virtual Machine (JVM)* zur Ausführung des compilierten Bytecodes der Programme existiert ([Sta07]). Die JVM dient dabei als Schnittstelle zur Maschine und zum Betriebssystem und ist für die meisten Plattformen verfügbar.

Bei *Java* können die Funktionalitäten eines Web Service in den Programmcode anderer Anwendungen integriert werden. Die Entwicklungsumgebung *JBuilderX* von der Firma *Borland* beispielsweise ermöglicht die Einbindung eines Web Service mit Hilfe eines *Web Service Designers* ([Bor07]). Hierbei wird die Beschreibung eines Web Service in Form eines WSDL-Dokumentes importiert, wodurch automatisch ein *Java-Package* generiert wird (in diesem Fall das Package `'simulator_control'`). In diesem Package befinden sich alle notwendigen Klassen und Methoden zur Einbindung eines Web Service in den Programmcode von *Java*. Das Package muss importiert und eine Instanz des Web Service unter Angabe einer URL (*Uniform Resource Locator*), eines Methodennamens und den Datentypen der Übergabeparameter erzeugt werden.

Die Logik und die zeitliche Abfolge der Web Services des in Abbildung 3.5 dargestellte Ablaufplan kann mit einem SEWS-spezifischen *Java-Programmcode* nachgebildet werden. Parallele Vorgänge sind in *Java* mit Hilfe von *Threads* realisierbar. *Threads* unterstützen so genannte nebenläufige Prozesse. Dabei versteht man unter der Nebenläufigkeit das gleichzeitige bzw. quasi-gleichzeitige Ablaufen von Aktivitäten auf einem Rechner ([Sta07]). Die überwiegende Anzahl an Programmiersprachen unterstützen solche Nebenläufigkeiten nicht oder nur unzureichend.

Die Verwaltung und Synchronisation der *Threads* wird mit Hilfe von *Thread-Groups* erleichtert ([Sta07]). So können bspw. alle *Threads* einer *Thread-Group* gleichzeitig gestoppt und gestartet werden. Weiterhin ist es möglich, mit der Verarbeitung des Programmcodes auf die Abarbeitung aller *Threads* einer *Thread-Group* zu warten.

Die Nachteile bei dieser spezifischen Anwendungslösung liegen in einem komplexen Ablaufprogramm, welches sowohl alle notwendigen Web Services einbinden als auch die vollständige Datenverwaltung und Flusskontrolle übernehmen muss. Die Programmierung von *Threads* stellt darüber hinaus hohe Anforderungen an den Entwickler. Weiterhin werden die Anforderungen Skalierbarkeit bzw. Adaptionfähigkeit, Erweiterbarkeit und Standardisierung bzw.

Wiederverwendbarkeit an SEWS dadurch nicht ausreichend unterstützt. Eine Möglichkeit, diese Nachteile zu vermeiden stellt die *Business Process Execution Language* dar, die im Folgenden näher erläutert wird.

3.5.2.2 Die Business Process Execution Language

Die Beschreibung des Workflows ist auch mit Hilfe spezieller Beschreibungssprachen möglich. Dies können beispielsweise Sprachen wie die *XML Process Definition Language (XPDL)* ([Sap06]), die *Business Process Modeling Language (BPML)* ([ADtHW07]), die *Web Services Choreography Description Language (WS-CDL)* ([W3C07b]) oder die *Business Process Modeling Notation (BPMN)* ([OMG07]) sein. Bei all diesen Beispielen handelt es sich um eine XML-basierte Metasprache zur Beschreibung von Geschäftsprozessen, deren einzelne Aktivitäten durch Web Services implementiert werden. Ein weiterer Vertreter dieser Sprachen ist die so genannte *Business Process Execution Language (BPEL)* ([Oas06]), die im Jahr 2002 von den Unternehmen *IBM*, *BEA* und *Microsoft* als Standard ins Leben gerufen wurde. In diesem Zusammenhang wird auch von der so genannten Web Service - Orchestrierung gesprochen, da eine Einbindung der mit WSDL beschriebenen Schnittstellen in den Code von BPEL erfolgt.

BPEL unterstützt nur die Interaktion zwischen Web Services. Die Kommunikation mit dem Menschen kann deshalb nur mit Hilfe eines Web Service geschehen. Zur Ausführung des Codes von BPEL wird eine so genannte *BPEL Engine* benötigt, die von den unterschiedlichsten Herstellern wie bspw. der *Oracle BPEL Process Manager* von *Oracle* ([Ora07]) oder die *BPWS4J* von *IBM* ([IBM07b]) angeboten werden.

Darüber hinaus existieren auch *Open Source*-Projekte zur Implementierung des BPEL-Standards wie *Twister* ([TP07]) oder *ActiveBPEL* ([Act07]). In einer BPEL-Engine können Fehler-, Kompensations- und Ereignisbehandlungen berücksichtigt werden. Der beschriebene BPEL-Prozess muss zusammen mit den WSDL-Informationen der verwendeten Engine zur Verfügung gestellt werden.

Zur Beschreibung des Workflows beinhaltet BPEL verschiedene Sprachbestandteile. Dies sind zum einen die so genannte *Basic Activities* zur Repräsentation atomarer Aktivitäten wie beispielsweise der Aufruf eines Web Service. Zum anderen existieren *Structured Activities*, die Kontrollstrukturen zur Anordnung der Basic Activities zur Verfügung stellen. Dadurch ist etwa eine sequentielle, bedingte oder parallele Durchführung von Aktivitäten möglich.

Ein Web Service wird durch seine WSDL-Datei beschrieben (vgl. Abschnitt 3.4.3). Diese Dateien sind die Voraussetzung für eine erfolgreiche Einbindung eines Web Services in BPEL. Jede BPEL-Datei besitzt grundsätzlich folgenden Grundaufbau ([JMS04]):

```
<process name='ProcessName' >
  <partnerLinks>..</partnerLinks>
  <variables>..</variables>
  <correlationSets>..</correlationSets>
  <faultHandlers>..</faultHandlers>
  <compensationHandler>..</compensationHandler>
```

```

        <eventHandlers>..</eventHandlers>
        activity
    </process>

```

In dem Bereich 'activity' kommen die Basic und Structured Activities zur Anwendung. Hier werden die eigentliche Ablaufreihenfolge und die Ablaufbedingungen des Geschäftsprozesses beschrieben. Soll hier ein Web Service aufgerufen werden, so muss dieser zunächst im Bereich der <partnerLinks> definiert werden.

Die Web Services *Datenquelle 1* und *Simulation Control* im Ablaufplan in Abbildung 3.5 können wie folgt eingebunden werden:

```

<partnerLinks>
    <partnerLinks name='sim_control'
        partnerLinkType='SimulatorControlLink'
        myRole='requestor' partnerRole='provider' />
    <partnerLinks name='datasourcel'
        partnerLinkType='DatasourcelLink'
        myRole='requestor' partnerRole='provider' />
</partnerLinks>

```

Die WSDL-Dateien dieser Web Services sind dann noch um die Spezifikation des so genannten <partnerLinkType> zu erweitern.

Dadurch sind die wesentlichen Informationen über die Kommunikationsbeziehungen der beteiligten Web Services und des Geschäftsprozesses beschrieben und können zur Abarbeitung während des Prozessablaufs instanziiert werden ([W3C07c]). Der Aufruf eines so eingebundenen Web Service erfolgt im 'activity'-Bereich des BPEL-Prozesses mit Hilfe der Basic Activity <invoke>, wobei der entsprechende Partner Link des Web Service, die gewünschte Methode und die notwendigen Inputparameter angegeben werden müssen.

Die Vorteile von BPEL liegen in der Verwendung von Standards wie SOAP, WSDL und XML. Dadurch wird die Entwicklung offener, flexibler und plattformunabhängiger verteilter Softwarearchitekturen ermöglicht. Eine synchrone und asynchrone Kommunikation unterstützt die Darstellung komplexer Geschäftsprozesse und zeitlicher Abläufe von Aktivitäten in Form von Web Services. Dabei wird ein Maximum an Flexibilität bei der Veränderung des Workflows und Erweiterung des Geschäftsprozesses um weitere Komponenten und Funktionalitäten erreicht.

Der Nachteil bei der Anwendung von BPEL liegt in der alleinigen Verwendung von Web Services. Jede Komponente, Berechnung, Funktion oder gar Kommunikation mit dem Menschen muss zunächst in Form eines Web Service zur Verfügung gestellt werden. In vielen Fällen ist dies nicht notwendig oder nicht wünschenswert, da bestimmte Funktionalitäten in Form von einfachem Programmcode bereits zur Verfügung stehen. Nicht zu vernachlässigen ist der bei einer reinen Verwendung von Web Services höhere Kommunikationsaufwand. Bei der Entwicklung von SEWS werden bestimmte Komponenten des Frameworks auf einem Master Web Server zur Verfügung gestellt (vgl. Abschnitt 3.4.3). Hierzu gehört auch das Data Management, welches Funktionalitäten zur SEWS-internen Datenverwaltung beinhaltet.

Im Rahmen dieser Arbeit gehört hierzu auch eine XML-Datenbank, in der die Daten in einem SEWS-spezifischen XML-Schema zusammengeführt und verwaltet werden. Je nach Datenbankoperation kann die Ansteuerung der XML-Datenbank sehr komplex sein. Dadurch ist die vollständige Kapselung dieser Ansteuerung mit Hilfe eines oder mehrerer Web Services für den Programmierer sehr unbequem. An dieser Stelle wäre es wünschenswert, verschiedene Funktionen in Form eines einfachen Programmcodes in BPEL einzubinden.

Dieses Problem kann mit Hilfe von *BPELJ (BPEL for Java)* gelöst werden ([IBM07a]). BPELJ ermöglicht die gemeinsame Verwendung von Java-Programmcodes und BPEL zur Darstellung eines Geschäftsprozesses. Hierbei können in BPEL Abschnitte mit Java-Programmcodes (so genannte *snippets*) definiert werden. Auch die Durchführung komplexer Operationen ist durch die Einbindung von Java-Packages möglich.

3.5.2.3 Zyklische Durchführung der Abläufe

Wie in Abschnitt 3.5.1 bereits angesprochen wurde, sind die Aktivitäten zyklisch durchzuführen. Sowohl eine spezifische Anwendungslösung als auch die Verwendung einer Beschreibungssprache benötigt letztendlich eine Art *Impuls*, der die Abarbeitung der Aktivitäten anstößt. Bei den EAI- und B2B-Ansätzen stellen solche Impulse bspw. Mitarbeiter, Kunden oder Partnerunternehmen dar. Aufgrund der Anforderungen Automatismus und Benutzerfreundlichkeit kann die zyklische Durchführung dem Anwender eines SEWS nicht zugemutet werden. Aus diesem Grund wurde für SEWS eine Anwendung entwickelt, welche eine so genannte 'pulse_function' darstellt und mit Hilfe der Angabe eines Zeitraums zwischen zwei Zyklen parametrisiert werden kann.

Auch hier bietet sich die Verwendung eines Threads an, der einen Zyklus startet und sich für den angegebenen Zeitraum bis zum nächsten Zyklus *schlafen* legt. Nachdem der Thread gestartet wurde, initialisiert und startet er die Aktivitäten eines SEWS. Dabei kann an dieser Stelle sowohl ein Aufruf des Kommunikationsstartpunktes eines BPEL-Prozesses oder der Start der Ablauflogik einer spezifischen Anwendungslösung erfolgen. Diese einfache Pulse-Funktion kann nun ebenfalls als Web-Anwendung auf einem Webserver implementiert werden.

3.6 Zusammenfassung des Kapitels

Dieses Kapitel beschäftigt sich mit der Thematik der Simulationsbasierten Frühwarnsysteme. Hierzu werden zunächst allgemeine Frühwarnsysteme anhand ihrer grundlegenden Funktionalitäten erläutert und auf strategische Frühwarnsysteme in Unternehmen eingegangen.

Der darauf folgende Abschnitt befasst sich speziell mit Simulationsbasierten Frühwarnsystemen in Produktion und Logistik. Eine große Variantenvielfalt und die verstärkte Kundenorientierung bewirken eine zunehmende Komplexität und Dynamik in der Produktion. Die Vorhersage künftiger Ereignisse, Zustände oder Entwicklungen ermöglicht einen zeitlichen Vorteil,

da frühzeitig geeignete Reaktionen oder Gegenmaßnahmen erfolgen können. Diese Prognosefunktionalität kann durch den Einsatz Simulationsbasierter Frühwarnsysteme erreicht werden. Allerdings existiert noch keine geeignete und umfassende Definition Simulationsbasierter Frühwarnsysteme und deren Abgrenzung in der Literatur. Aus diesem Grund wird eine Definition hergeleitet und Anforderungen an Simulationsbasierte Frühwarnsysteme erläutert, die mit den allgemeinen Funktionalitäten eines Frühwarnsystems verglichen werden. Im Anschluss erfolgt eine Abgrenzung und Einordnung Simulationsbasierter Frühwarnsysteme im Vergleich zu anderen simulationsbasierten Ansätzen.

Der daran anschließende Abschnitt befasst sich mit einer allgemeinen Architektur Simulationsbasierter Frühwarnsysteme. Hierzu werden zunächst die Hauptkomponenten Datenquellen, Simulationsmodelle, Anwender und Framework vorgestellt. Das Framework übernimmt dabei vielfältige Funktionalitäten. Diese ergeben sich zum einen aus den Anforderungen an Simulationsbasierte Frühwarnsysteme und zum anderen aus der notwendigen Integration der Hauptkomponenten zu einer Gesamtarchitektur. Die Komponenten des Frameworks werden anschließend hergeleitet und näher erläutert.

In einem weiteren Abschnitt geht es um verteilte Systeme und Kommunikationsprinzipien, die bei der Entwicklung einer verteilten Architektur Simulationsbasierter Frühwarnsysteme anwendbar sind. Hierbei werden zunächst die Technologien RPC, RMI, CORBA und DCOM hinsichtlich ihrer Vor- und Nachteile untersucht. Anschließend werden Web Services als geeignete Technologie zur Entwicklung eines verteilten Systems identifiziert und im Zusammenhang mit Simulationsbasierten Frühwarnsystemen betrachtet.

Zum Abschluss wird die Steuerung des Workflows Simulationsbasierter Frühwarnsysteme untersucht. Hierzu werden die Interaktionen der Komponenten eines Simulationsbasierten Frühwarnsystems in Form eines Ablaufplans dargestellt. Weiterhin erfolgt eine Vorstellung von Steuerungsmöglichkeiten für den Workflow, die im Rahmen dieser Arbeit aus einer spezifischen Anwendungslösung und der Business Process Execution Language bestehen. Abschließend wird auf die Thematik der zyklischen Durchführung der Abläufe eingegangen.

Dieses Kapitel erläutert die allgemeinen Grundlagen und Zusammenhänge Simulationsbasierter Frühwarnsysteme. Dabei werden viele Fragen aufgeworfen, die in den folgenden Kapitel beantwortet werden sollen. So stellt sich etwa die Frage, wie die Anbindung heterogener Datenquellen erreicht und eine effiziente Datenverwaltung für Simulationsbasierte Frühwarnsysteme aussehen kann. In diesem Zusammenhang ist zu klären, welche Rolle XML spielt und hinsichtlich der gestellten Anforderungen an ein SEWS unterstützen kann.

Weiterhin ist die Einbindung unterschiedlicher Simulatoren zu erläutern, so dass der Anforderung der Unabhängigkeit Rechnung getragen wird. Dabei sind besonders der notwendige Modellaufbau, in das Simulationsmodell zu implementierende Module und die Initialisierung von Simulationsmodellen interessant. Weiterhin stellt sich die Frage, wie Ausnahmefälle detektiert werden können. Darüber hinaus sind die Anforderungen Adaptionfähigkeit und Skalierbarkeit hinsichtlich ihrer Auswirkungen auf die Prognosegenauigkeit zu diskutieren.

Ein weiteres komplexes Themengebiet stellt die Generierung von Handlungsalternativen dar. Hierzu ist zu diskutieren, wie die notwendigen Komponenten und Funktionalitäten aussehen müssen, um Simulationsbasierte Frühwarnsysteme möglichst performant zu gestalten.

Und schließlich ist sowohl die technische Realisierung und Integration in einer Unternehmensinfrastruktur als auch die praktische Umsetzung anhand von Fallbeispielen von Interesse.

Kapitel 4

Prognose künftiger Zustände des realen Systems

*Vorhersagen sind immer schwierig
- vor allem über die Zukunft.
— Niels Bohr*

In diesem Kapitel wird die Prognose künftiger Zustände des realen Systems mit Hilfe der Simulation erläutert. Ein erster Abschnitt beschäftigt sich mit der Datenbeschaffung und Datenverwaltung in Simulationsbasierten Frühwarnsystemen (SEWS) im Bereich von Produktion und Logistik. Dabei werden zunächst simulations- und SEWS-relevante Daten unterschieden, die in den Informationssystemen heutiger Industrieunternehmen zu finden sind. Anschließend werden mögliche Kriterien zur Systemauswahl bei der Datenbeschaffung betrachtet, wobei eine Darstellung der SEWS-internen Datenverwaltung mit Hilfe von XML erfolgt. Weiterhin wird die Verwendung einer XML-Datenbank begründet. Der Abschnitt schließt mit der Erläuterung der Datengewinnung aus originären Datenquellen. Ein weiterer Abschnitt befasst sich mit der Simulation als eingebettete Applikation. Dabei werden sowohl auf relevante Eigenschaften von Simulatoren als auch die Anforderungen an Simulationsmodelle zur Verwendungen in einem SEWS eingegangen. Das Kapitel endet mit einem Abschnitt über die Rolle der Simulation bei der Vorhersage künftiger Zustände des realen Systems. Es werden wichtige Themenschwerpunkte wie die Initialisierung von Simulationsmodellen, die Prognose von Ausnahmefällen, Möglichkeiten zur Modellaktualisierung und das Problem der Prognosegenauigkeit diskutiert.

4.1 Datenbeschaffung und -verwaltung

Sowohl zur Durchführung von Simulationsstudien in Produktion und Logistik als auch der Einsatz von Simulationsmodellen in einem SEWS sind abhängig von Daten über das reale oder geplante System. Die Kopplung von Informationssystemen mit der Simulation bzw. Simulationsmodellen ist ein Themengebiet, welches in zahlreichen Arbeiten diskutiert wird und eine hohe Relevanz für die Praxis besitzt ([LL04],[RB01],[Ber00],[GRS02]). Dieser Abschnitt geht zunächst darauf ein, welche Daten simulations- bzw. SEWS-relevant sind. Anschließend erfolgt eine Darstellung von Kriterien zur Systemauswahl, wobei sowohl Themen wie Performance bei der Datenbeschaffung, Datenaktualität und -qualität als auch die Synchronität der Systeme diskutiert werden.

4.1.1 Simulations- und SEWS-relevante Daten

Die zur Durchführung von Simulationsstudien notwendigen Daten sind sehr stark abhängig von der verfolgten Zielsetzung und den in den Modellen verwendeten bzw. vorhandenen Simulationsobjekten. Bei der Anwendung der Simulation in Produktion und Logistik können grundsätzlich alle Daten relevant sein, die in industriellen DV-Systemen vorhanden sind. Reinhardt et al. beschreibt solche simulationsrelevante Daten, bei denen es sich um Daten zu Prozessen, dem Produktionsprogramm, dem Werkzeugmanagement, dem Routing, der Qualitätskontrolle und dem Werkereinsatz handelt (vgl. hierzu Abb. 4.1) ([RNJ03]).

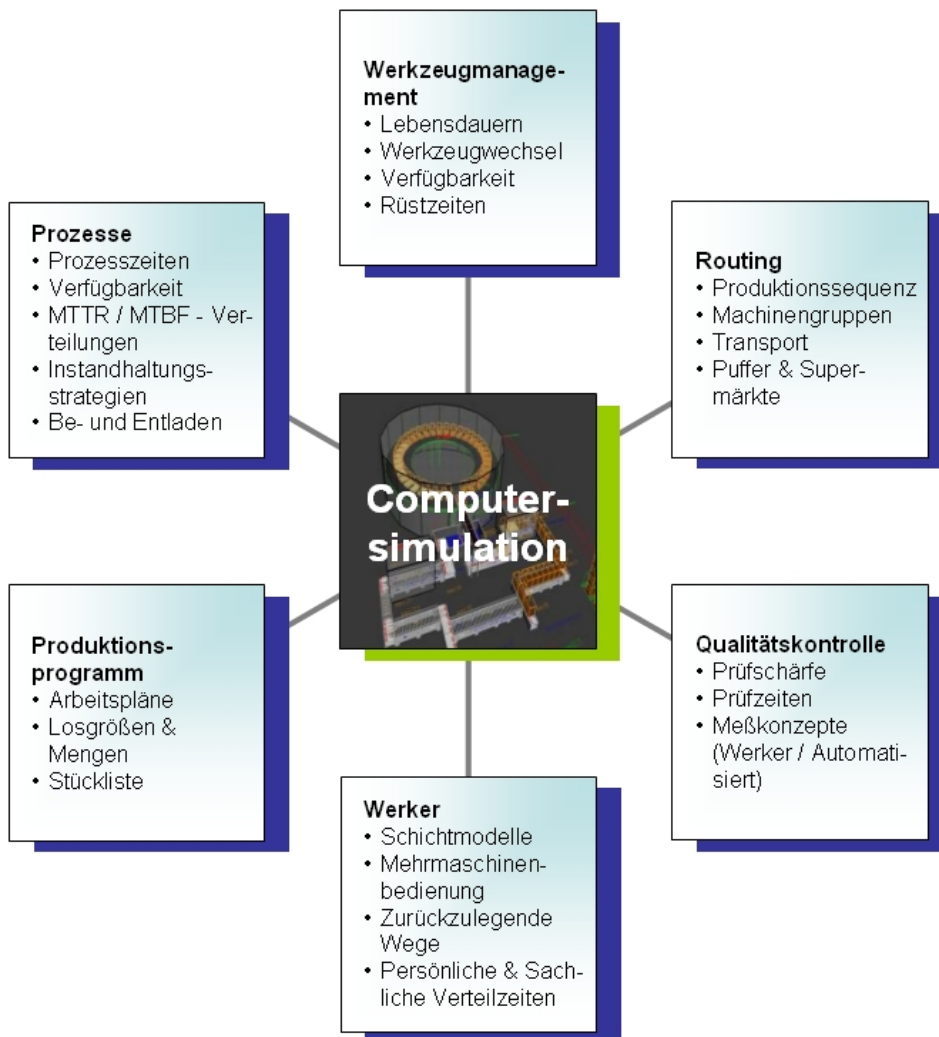


Abb. 4.1: Simulationsrelevante Daten im traditionellen Sinne ([RNJ03])

Dieser traditionelle Datenumfang ist für SEWS nicht ausreichend. Für simulationsrelevante Daten im Sinne von SEWS ergibt sich ein entscheidender Mehrbedarf aus der Notwendigkeit der Initialisierung des Simulationsmodells mit dem aktuellen Zustand des realen Systems und der Konfiguration eines Simulationsbasierten Frühwarnsystems.

Bei diesen additiven Datenumfängen handelt es sich um *dynamische* Informationen, die von den bisher erläuterten (Anm.: Und folglich komplementär als *statisch* aufzufassenden) Datenumfängen abzugrenzen sind. Die Unterscheidung dieser Charakterisierung liegt im Zeitbezug der Daten zur Durchführung von Prognosen begründet. Statische Informationen verändern sich über länger Zeiträume nicht, wohingegen dynamische Daten von Prognosezyklus zu Prognosezyklus unterschiedlich sind (vgl. Abschnitt 3.5.2.3). Es kann folgende Klassifikation getroffen werden:

- **Ressourcenstatus**

Eine weitere notwendige Information zum Zeitpunkt der Modellinitialisierung ist der Status einzelner Ressourcen im Materialflusssystem. Beispiele für Ressourcen sind Werker, Werkzeuge und Maschinen. Der Werkerstatus ist typischerweise abhängig vom Schichtverlauf. Werkzeuge können sich gerade in Reparatur befinden oder es wird ein Werkzeugwechsel durchgeführt. In Bezug auf Maschinen sind Rüst-,Warte- und Bearbeitungsstati zu berücksichtigen. Weiterhin ist es erforderlich, über die Statusinformation hinaus noch den Zeitstempel zu erfassen, zu dem die letzte Statusänderung stattgefunden hat. Dadurch können bspw. Restbearbeitungszeiten, der Fortschritt bei Rüstzeiten und die restlichen Pausenzeiten ermittelt werden.

- **Status der Produkte**

Der Status der Produkte ist mit Hilfe der Messpunkte feststellbar. Messpunkte sind Messstellen in dem realen System, an denen durch automatische Sensormessungen oder manuelle Rückmeldepunkte Veränderungen von Produktstati registriert und in den zugrunde liegenden Informationssystemen hinterlegt werden. Jedes Produkt wird an einem Messpunkt mit entsprechendem Zeitstempel erfasst, sobald es diesen passiert. Dadurch kann der aktuelle lokale Standort eines Produktes im Materialflusssystem ermittelt werden. Zwischen diesen Messpunkten ist die Verfolgung der Produkte im Allgemeinen nicht möglich. Je dichter die Messpunkte in einem Materialflusssystem verteilt sind, desto kleiner sind die Initialisierungsfehler beim Start des Simulationsmodells. Dies wirkt sich direkt auf die Prognosegüte der Simulationsläufe aus.

Neben den simulationsrelevanten Daten sind noch SEWS-relevante Daten wie die Beschreibung von Ausnahmefällen, sowie Konfigurations- und Historiedaten von Bedeutung. Zur Definition von Ausnahmefällen muss eine geeignete Darstellungsform gewählt werden, die sowohl von einem Simulationsmodell als auch vom Anwender interpretiert werden kann. Historiedaten umfassen Informationen zu vergangenen Simulationsläufen, etwaigen Ausnahmefällen und generierten Handlungsalternativen. Zu den Konfigurationsdaten gehören Parameter zur Ansteuerung der Simulationsmodelle und Informationen über die abzufragenden DV-Systeme des realen Systems.

Eine weitere Unterscheidung der simulationsrelevanten Daten kann hinsichtlich ihrer Relevanz für die Prognose getroffen werden, bei denen es sich im Folgenden um drei mögliche Arten handelt.

- **Prognoserelevante Daten ersten Grades**

Diese Daten entsprechen den dynamischen Daten, die für jede Prognose aktualisiert wer-

den müssen, da sie den gegenwärtigen Zustand des realen Systems beschreiben. Auch die Informationen über die Belegungsplanung müssen ständig aktualisiert werden.

- **Prognoserelevante Daten zweiten Grades**

Die statischen Daten verhalten sich unabhängig von einem spezifischen Systemzustand. Dennoch können sie sich über längere Zeiträume hinweg verändern. Dies betrifft bei Produktionsanlagen bspw. Prozesszeiten, wie etwa manuelle Bearbeitungsdauern. Vor allen Dingen bei einem großen Anteil an manuellen Tätigkeiten verringern sich durch mehr Übung des Personals die Durchlaufzeiten. Weiterhin werden ständig Optimierungen im Materialflusssystem vorgenommen, so dass sich die Eingangsdaten der Simulation verändern können.

- **Nicht prognoserelevante Daten**

Im Gegensatz zu den ersten beiden Datengruppen verändert sich der grundlegende Aufbau wie der Standort oder die Anzahl von Maschinen, Verzweigungen und Zusammenführungen des Materialflusssystem selten ([AF05]). Diese Daten sind für einen längerfristigen Betrachtungszeitraum von Bedeutung. Hierbei ist eine vollständige Automatisierung der Modellaktualisierung anzuzweifeln, da neue Produkte, neue Maschinen oder veränderte Technologien hohe Freiheitsgrade bei den möglichen Veränderungen von Materialflusssystemen bewirken. Aus diesem Grund sollen diese Daten in diese Arbeit keine größere Rolle spielen und werden bei der Modellentwicklung implizit berücksichtigt.

Eine Ausnahme bilden künftige Belegungs- und Produktionsplanungsdaten, durch die das Modell ausgehend vom gegenwärtigen Zustand des realen Systems Prognosen treffen kann. Es handelt sich hierbei um statische Daten, die dennoch zu den Prognoserelevanten Daten ersten Grades gehören. Eine zusätzliche Erweiterung der bisherigen Klassifikation wird durch die Einführung von Eingangsdaten zur Repräsentation des gegenwärtigen Systemzustandes und Plandaten für den Prognosezeitraum erreicht. Abbildung 4.2 verdeutlicht diese Zusammenhänge nochmals.

In großen Industrieunternehmen finden sich meist historisch gewachsene IT-Systeme, die den technischen Fortschritt der Informationstechnologie im Allgemeinen bzw. der Datenbanktechnologie im Speziellen widerspiegeln. Es kommen neue IT-Systeme mit neuen Funktionalitäten und Möglichkeiten hinzu, die in die bestehende Systemwelt integriert werden müssen. Während zu Beginn des Einsatzes von Produktions- und Informationssystemen in der Industrie monolithische Datenbankmanagementsysteme zum Einsatz gekommen sind, sind dies heutzutage aufgrund verbesserter und standardisierter Schnittstellentechnologien wie bspw. XML und der größeren Anzahl von Systemanbietern vermehrt kleinere, spezialisierte und autonome IT-Systeme.

Bei der Implementierung eines SEWS muss aus dieser Vielfalt von inhomogenen Informationssystemen eine Auswahl derjenigen IT-Systeme erfolgen, aus denen die zur Initialisierung der Simulationsmodelle notwendigen relevanten Daten ermittelt werden können.

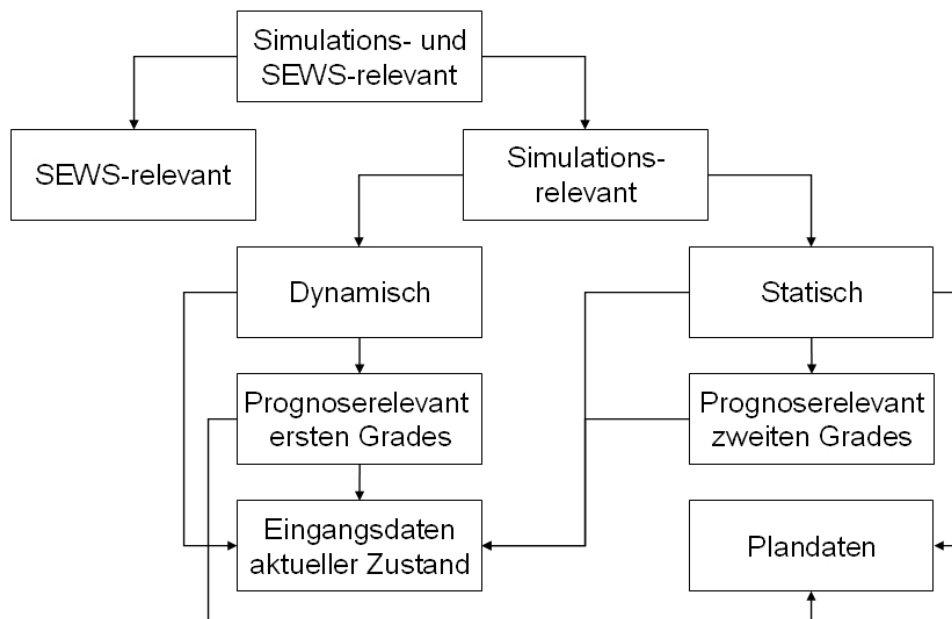


Abb. 4.2: Klassifikation der simulations- und SEWS-relevanten Daten

4.1.2 Kriterien zur Systemauswahl

Die Literatur zur Kopplung der existierenden Datenwelt mit der Simulation beschränkt sich vielfach auf die technische Umsetzung und die Beschreibung der simulationsrelevanten Daten ([GRS02],[Ber00],[Joh01]). Dabei wird stillschweigend davon ausgegangen, dass entweder alle Daten in einem monolithischen *Enterprise Resource Planning (ERP)* - System vorhanden sind, oder dass die Daten über diverse Schnittstellen in einer Simulationsdatenbank zusammengeführt werden können. Es fehlt eine kritische Auseinandersetzung mit der bestehenden Heterogenität der IT-Systeme in Industrieunternehmen und dem dadurch entstehenden Zwang, eine Systemauswahl zu treffen. Im Folgenden sollen mögliche Kriterien diskutiert werden, nach denen eine solche Auswahl erfolgen kann.

- **Datenverfügbarkeit**

Die Nutzung eines IT-Systems für SEWS ist nur dann interessant, wenn es simulationsrelevante Daten enthält. Eine Ausnahme stellen IT-Systeme dar, aus denen zunächst Schlüsselinformationen (bspw. *Metadaten*, vgl. [Sta02]) erforderlich sind, um relevante Daten aus einem anderen IT-System erhalten zu können.

- **Datenaktualität und Datenqualität**

Weitere Kriterien stellen die Aktualität und Qualität der Daten dar. Dabei ist unter der Datenqualität die Bedeutsamkeit, Relevanz und Korrektheit von Daten zu verstehen, also wie gut Daten in der Lage sind, die Realität zu beschreiben. Somit ist die Datenqualität ein Maß dafür, wie verlässlich eine Information ist und inwieweit sie als Grundlage für Planungen des eigenen Handelns dienen kann ([Epp06]). Unter der Datenaktualität ist die Zeitnähe von Daten im Zusammenhang mit dem gegenwärtigen Zeitpunkt zu verstehen.

Stehen Daten in mehreren IT-Systemen redundant zur Verfügung, so ist das IT-System auszuwählen, das die gewünschten Daten mit der höchsten Qualität und Aktualität liefert. Eine ausreichende Datenqualität ist das wichtigste Kriterium für die Anwendung eines SEWS, da bspw. die Prognosegüte und die Generierung von Handlungsalternativen davon abhängig sind. Aus diesem Grund ist zuvor die Datenqualität der verwendbaren IT-Systeme genauestens auf ihre Eignung hin zu prüfen.

- **Performance**

Es ist wichtig, dass die gewünschten Informationen in kurzer Zeit gewonnen werden können. Auch hier sind die IT-Systeme mit der größten Performance auszuwählen.

- **Abfrage- und Anbindungskomplexität**

Die Komplexität bei der Anbindung und bei der Gestaltung von Abfragen zur Datengewinnung aus den verwendeten IT-Systemen ist ebenfalls von Bedeutung bei der Systemauswahl. Unabhängig von der Performance wird dadurch der Implementierungsaufwand von SEWS beeinflusst, weshalb bei vorhandener Datenredundanz grundsätzlich diejenigen IT-Systeme mit der geringsten Abfrage- und Anbindungskomplexität auszuwählen sind.

- **Synchronität der Systeme**

Jedes IT-System wird entweder kontinuierlich oder zu spezifischen diskreten Zeitpunkten mit dem gegenwärtigen Zustand des Materialflusssystems aktualisiert. Vor dieser Aktualisierung repräsentieren die IT-Systeme vergangene Systemzustände. Sind simulationsrelevante Daten über mehrere IT-Systeme verteilt, führen diese Synchronisationszeitpunkte und die dazwischen liegenden Zeitintervalle zu Zeiträumen, in denen die IT-Systeme unterschiedliche Zustandsaktualitäten repräsentieren. Aus diesem Grund müssen alle ausgewählten IT-Systeme die Voraussetzung der Synchronität erfüllen.

- **Nichttechnische Rahmenbedingungen**

Bei der Auswahl von geeigneten IT-Systemen zur Einbindung in ein SEWS sind weiterhin nichttechnische Rahmenbedingungen von Bedeutung. Selbst IT-Systeme, die nach den vorangegangenen Kriterien für geeignet befunden werden können, sind aus diversen Gründen nicht verwendbar. Solche Gründe können bspw. vertragliche Vereinbarungen zwischen Systemanbietern und Industrieunternehmen oder arbeitsrechtliche Bedenken sein.

Die Auswahl der IT-Systeme ist im Einzelfall zu prüfen, da auch anhand der hier vorgestellten Kriterien keine scharfe Abgrenzung von Prioritäten oder eine globale Gewichtung der Kriterien möglich ist. Allerdings sind allgemeine Zusammenhänge ableitbar, die im Folgenden anhand der Parameterpaarung Datenredundanz und Priorität des Kriteriums erläutert werden.

Im Gegensatz zur Datenverfügbarkeit nimmt die Priorität der Anbindungs- und Abfragekomplexität mit zunehmender Datenredundanz zu. Gleiches gilt für die Kriterien der Performance, der Synchronität, sowie der Datenaktualität und -qualität, die von großer Bedeutung für die Anforderungen an ein SEWS sind (vgl. Abschnitt 3.2.2). Die Priorität der Performance ist höher einzuschätzen, da sie auch ein Ergebnis der Synchronität und Datenaktualität bzw. -qualität ist. So ist es möglich, dass eine niedrigere Performance bei der Datenbeschaffung aus

einem IT-System dazu führt, dass ein anderes IT-System mit geringerer Datenaktualität aber höherer Performance bei der Datenbeschaffung ausgewählt wird. Allerdings haben sowohl die Datenverfügbarkeit, Datenaktualität und -qualität große Auswirkungen auf die Initialisierung der Simulationsmodelle mit dem gegenwärtigen Zustand des Materialflusssystem und auf die Prognose künftiger Systemzustände. Die nichttechnischen Rahmenbedingungen sind unabhängig von der Datenredundanz und können mit einer mittleren Priorisierung versehen werden, da sie entweder eine Rolle spielen können oder nicht. Abbildung 4.3 visualisiert diese Zusammenhänge noch einmal.

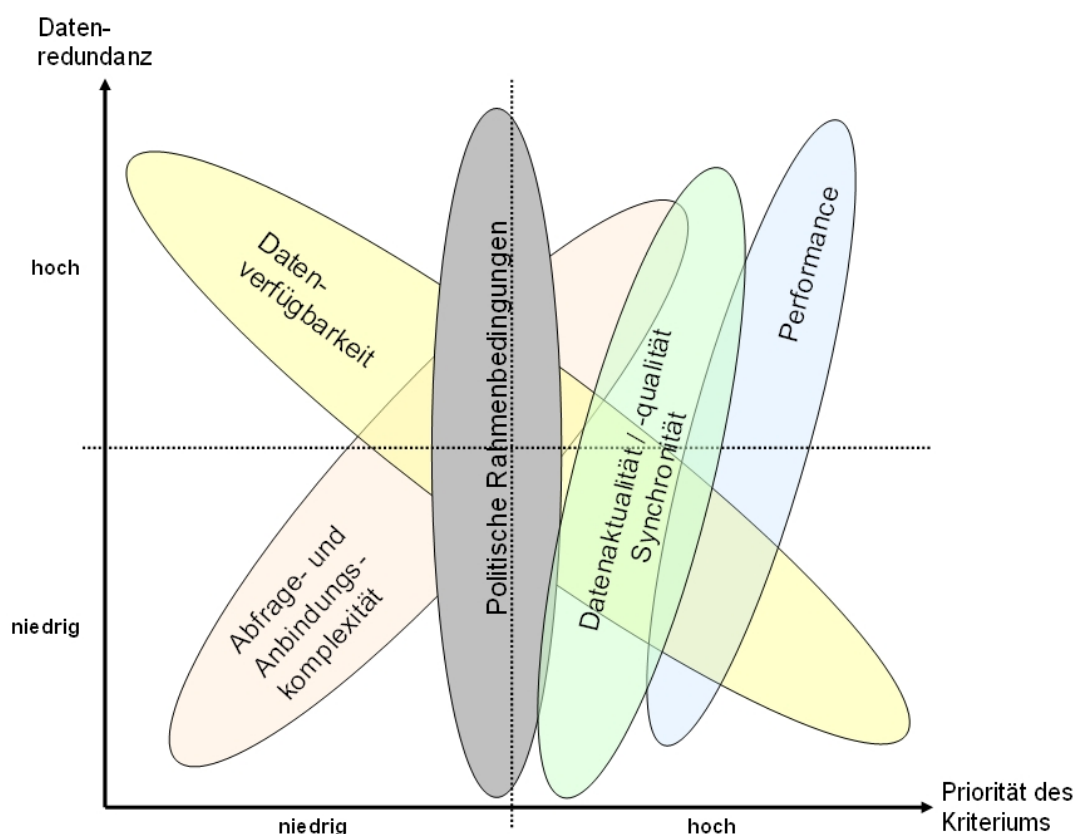


Abb. 4.3: Zusammenhang der Kriterien zur Auswahl von IT-Systemen

Sind bestimmte Datenumfänge nur in einem IT-System enthalten, so gibt es unabhängig von den hier aufgeführten Kriterien keine Alternativen. Ist weiterhin eine ausreichende Datenqualität nicht gewährleistet, so ist der Einsatz von SEWS nicht sinnvoll. Im folgenden Abschnitt wird die SEWS-interne Datenverwaltung mit Hilfe von XML erläutert.

4.1.3 SEWS-interne Datenverwaltung mit XML

Zur Unterstützung der Anforderung Unabhängigkeit (vgl. Abschnitt 3.3) wird XML zur SEWS-internen Datenverwaltung herangezogen. Dadurch ist eine übersichtliche Beschreibung

aller notwendigen Daten für ein SEWS in Form einer XML-basierten Datenstruktur möglich.

In der Literatur existieren bereits Ansätze zur Beschreibung von Simulationsdaten mit Hilfe von XML. Lu et al. stellt eine prototypische Spezifikation zum Datenaustausch wiederverwendbarer Simulationsdaten im Montagebereich eines Flugzeugherstellers vor ([LQM03]). Einen ähnlichen Ansatz beschreibt McLean et al., der eine XML-basierte Datenstruktur zum Datenaustausch von Fertigungsinformationen zwischen Simulationsanwendungen und anderen Fertigungsapplikationen oder Datenquellen ermöglicht ([MJLR02]). Auch zur Spezifikation von Simulationsmodellen wird vermehrt XML eingesetzt, um bspw. eine simulatorunabhängige Modellierung und Anpassung von Modellen zu erleichtern ([Fis02]). Wiedemann verwendet XML-basierte Codegeneratoren, um Simulationsmodelle zwischen verschiedenen Laufzeitumgebungen zu konvertieren ([Wie05]).

Die Vielfalt solcher Ansätze zeigt, dass gegenwärtig keine globale Spezifikation zur Beschreibung von Simulationsmodellen oder simulationsrelevanten Daten in XML existiert. Dies liegt an der Vielzahl von Anwendungsmöglichkeiten zur Datenbeschreibung mit Hilfe von XML, welche höchst unterschiedliche Anforderungen an eine Datenstruktur stellen. Aus diesem Grund wird im Rahmen dieser Arbeit eine für SEWS global gültige Datenstruktur entwickelt, die auf unterschiedliche Problemstellungen angewendet werden kann.

Hierzu sind Schema-Definitions-Formate wie die *Document Type Definition (DTD)* oder die *XML Schema Definition (XSD)* geeignet ([EE04],[Sku04]). XSD ist der mächtigere Nachfolger von DTD zur Beschreibung von XML-Dokumenten. Dies liegt an der umfangreichen Unterstützung von Datentypen und der detaillierteren Darstellung von Nebenbedingungen hinsichtlich der logischen Struktur eines XML-Dokumentes.

Abbildung 4.4 zeigt das globale XML-Schema (XSD) für die vollständige Datenverwaltung in SEWS, das im Rahmen dieser Arbeit entwickelt wurde. Es handelt sich hierbei um eine schematische Darstellung mit Hilfe des *XML-Viewers XMLSpy* der Firma *Altova* ([Alt06]). Den Ursprungsknoten des XML-Schemas bildet das komplexe Element `<ProductionSystem>` und dient als Repräsentation für das mit einem SEWS auszustattende reale System. Es enthält weitere komplexe und einfache Kind-Elemente zur Beschreibung der hierzu notwendigen Daten.

Die komplexen Kind-Elemente `<ProcessSystem>`, `<Resources>`, `<ShiftModels>`, `<MeasuringPoints>`, `<ProcessSequences>`, `<Products>` und `<WorkOrderList>` dienen zur Beschreibung des realen Materialflusssystems. Darin sind sowohl die dynamischen Daten über den aktuellen Systemzustand als auch vom gegenwärtigen Systemzustand unabhängige Daten enthalten. Dabei handelt es sich bspw. um Prozesszeiten, Prozesssequenzen, allgemeine Ressourceninformationen und eine Liste der Messpunkte (vgl. Abschnitt 4.1.1).

Zur Beschreibung von möglichen Ausnahmefällen zur späteren Ausnahmefalldetektion finden die Elemente `<Exceptions>` und `<Rules>` Anwendung. Das Element `<Simulation>` dient zur Konfiguration von Steuerparametern des SEWS wie bspw. die Festlegung eines Zeitraums zwischen der zyklischen Wiederholung einer Prognose (vgl. Abschnitt 3.5.2.3). Zur

Verwaltung der Simulationsergebnisse wird das komplexe Kind-Element <Results> verwendet.

Die einfachen Kind-Elemente <SystemFile>, <Description> und <ProductionSystemID> dienen zur Verwaltung der systemspezifischen SEWS-Daten. Eine genauere Erläuterung und Darstellung der einzelnen Elemente findet sich im Anhang A.

Mit Hilfe dieses globalen XML-Schemas können alle Simulations- und SEWS-relevanten Daten verwaltet werden. Die Anbindung weiterer Applikationen oder Informationssysteme wird erleichtert, die auf die Daten eines SEWS zugreifen sollen. Dies können künftig bspw. Visualisierungs- oder Auswertetools sein, die das globale XML-Schema verwenden können, obwohl sie nur Teilinformationen aus dem gesamten Datenumfang eines SEWS benötigen. Dadurch erfolgt eine konsequente Unterstützung der Anforderungen Unabhängigkeit, Erweiterbarkeit, Wiederverwendbarkeit, Skalierbarkeit und Adaption.

Eine Möglichkeit zur Verwaltung von XML-Dokumenten besteht darin, diese in einem lokalen oder in einem Netzwerk vorhandenen Ordner abzuspeichern. Nachteilig bei dieser Lösung ist die schwierige Zugriffskontrolle. Weiterhin sind sicherheitsrelevante Fragestellungen zu lösen, die bei einer rein lokalen Datenhaltung auftreten. Die effizientere Möglichkeit, XML-Dokumente und die darin enthaltenen Informationen zu verwalten, stellen XML-Datenbanken dar.

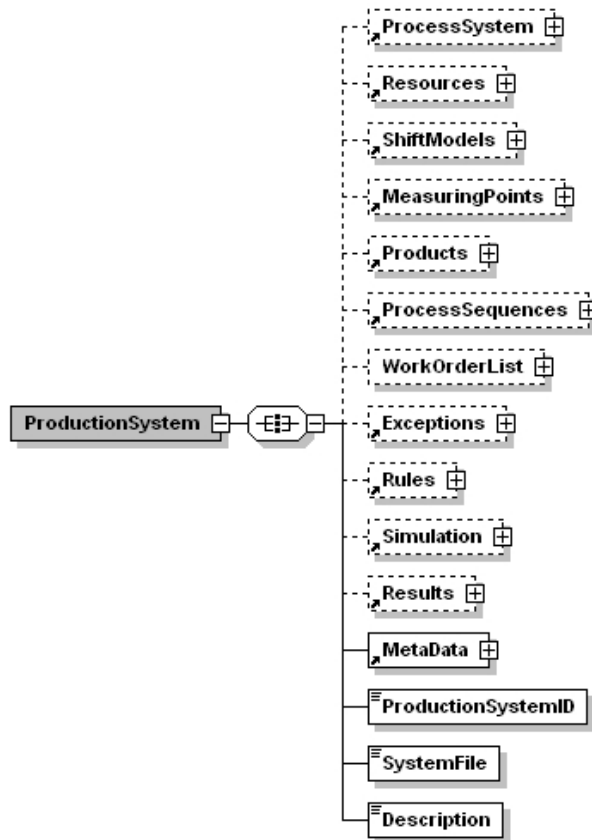


Abb. 4.4: XML-Schema für Simulationsbasierte Frühwarnsysteme

4.1.4 Verwaltung mit Hilfe einer XML-Datenbank

Datenbanken oder Datenbankmanagementsysteme, die Daten im XML-Format speichern oder verarbeiten können, werden als *XML-Datenbanken* bezeichnet. Hierbei sind *native XML-Datenbankensysteme* und Datenbanksysteme, die *XML-enabled* sind, zu unterscheiden ([Cha03]). Letztere sind herkömmliche Datenbanken, die ein Mapping von Daten in das XML-Format erlauben.

Native XML-Datenbanken arbeiten dokumentorientiert und speichern XML-Dokumente direkt ab, wobei eine Indizierung der Elemente des XML-Dokumentes erfolgt. Der Vorteil von nativen XML-Datenbanken liegt in dem fehlenden Transformationsprozess aus bzw. in das XML-Format.

Im vorangehenden Abschnitt wird eine XML-Struktur zur SEWS-internen Datenverwaltung vorgestellt. Hierin sind auch die zur Initialisierung von Simulationsmodellen notwendigen Daten enthalten. Diese Daten im XML-Format müssen in einer effizienten Art und Weise zusammengeführt und verwaltet werden. An dieser Stelle ist die Frage zu beantworten, warum sich eine native XML-Datenbank zum Einsatz in SEWS anbietet und nicht etwa relationale oder objektorientierte Datenbanken verwendet werden.

XML-Dokumente stellen eine Baumstruktur dar. Relationale Datenbanken dagegen verwalten Daten in Tabellen oder in einer matrixartigen Struktur und verwenden relationale Verknüpfungen zur hierarchischen Darstellung der Datenzusammenhänge. Es ist sehr einfach, relationale Daten in XML zu überführen. Die umgekehrte Richtung, also XML in relationalen Datenbanken abzubilden, kann sehr schwierig sein und zu Informationsverlusten führen.

Objektorientierte Datenbanken sind sehr stark mit den Applikationen verbunden, die diese verwenden. Sie sind im Allgemeinen nicht dazu bestimmt, applikationsunabhängige Datenbanken zu sein ([MW03]). Üblicherweise können die von solchen Datenbanken verwalteten Objekte nur mit Hilfe derjenigen Programmiersprache verwendet werden, die diese auch erzeugt haben. Darüber hinaus führt eine große Anzahl von Objekten, die durch die Verwaltung großer Sammlungen von XML-Dokumenten entstanden sind, zu einem ernsthaften Performance- und Ressourcenengpass.

Aus diesen Gründen wird im Rahmen dieser Arbeit *Xindice* verwendet ([Apa07]). Es handelt sich dabei um eine native *open source* XML-Datenbank, die komprimierte XML-Dokumente speichert und indiziert, um diese Daten unterschiedlichen Applikationen zur Verfügung zu stellen. Darüber hinaus umfasst sie im Gegensatz zu relationalen Datenbanken Funktionalitäten, die den Umgang mit XML und seinen Eigenarten unterstützen.

Zur Datenbankabfrage wird die von W3C spezifizierte Abfragesprache *XML Path (XPath)* verwendet, die zur Knotenauswahl von Dokumenten in einer XML-Dokumentensammlung dient ([WL03]). Ein XPath-Ausdruck setzt sich aus einem oder mehreren *Lokalisierungsschritten* zusammen, mit denen in der Baumstruktur eines XML-Dokumentes *navigiert* wird. Die XML-Datenbank kann diese Lokalisierungsschritte nachvollziehen und stellt das Ergebnis anschließend im XML-Format zur Verfügung.

Die Manipulation von Daten in einer XML-Datenbank erfolgt mit Hilfe von *XML Update (XUpdate)* ([WL03]). Es handelt sich dabei um eine XML-basierte Beschreibungssprache, wobei XPath implizit zur bedingten Selektion Verwendung findet. Sie ermöglicht eine effiziente Erzeugung, Aktualisierung und Löschung von Datensätzen, wie sie bspw. in relationalen Datenbanksystemen vorhanden ist.

Die für SEWS in XPath und XUpdate definierten Befehle können sehr aufwändig gestaltet sein. Um eine effiziente Verwaltung zu ermöglichen, wurde ein XML-Schema zur Definition von XPath- und XUpdate-Befehlen entwickelt (vgl. Abb. 4.5). Dadurch ist eine schnelle und flexible Konfiguration von XML-Datenbankoperationen für verschiedene Applikationen möglich.

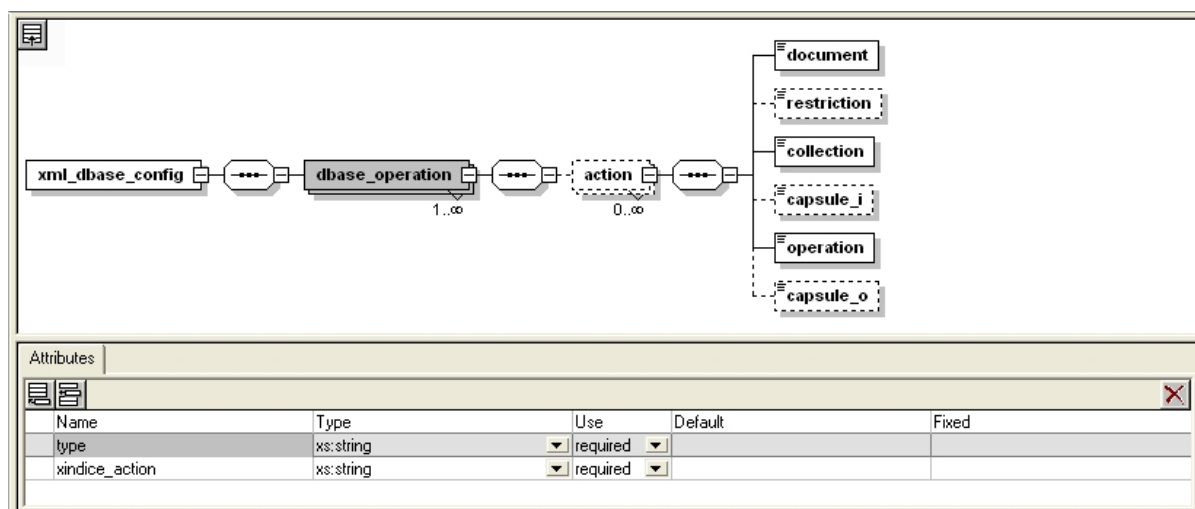


Abb. 4.5: XML-Schema zur Verwaltung von XPath- und XUpdate-Befehlen

4.1.5 Datengewinnung aus originären Datenquellen

Zur realitätsgetreuen Abbildung bestehender Materialflusssysteme wird eine Vielzahl unterschiedlicher Informationen benötigt, die sich in unterschiedlichen Datenbanken eines produzierenden Unternehmens befinden. Diese IT-Infrastruktur ist meist historisch über die Jahre gewachsen und es finden sich Datenbanksysteme verschiedenster Hersteller und Architektur.

Zur Integration von SEWS in eine bestehende IT-Welt, müssen die Datenbanksysteme inhaltlich zusammengeführt werden, die SEWS- und simulationsrelevante Daten enthalten. Diese Datenbanken sind meist als autonome Einzelsysteme konzipiert und sehen keine technische Kopplung vor. Eine weitere Herausforderung bei der Integration stellen die teils unterschiedlichen Formate der Inhalte dar. Zwar lassen sich über Schlüsselfelder Daten zusammenführen, jedoch trifft man häufig auf verschiedene Merkmalsausprägungen. Hierzu zählen unterschiedliche Datentypen und Darstellungsformen.

Die Verwendung einer XML-Datenbank entspricht dem Ansatz einer datenbankgestützten Simulation. Hierbei werden die Daten aus den originären Datenbanksystemen mit unterschiedli-

chen Aufbauarchitekturen gewonnen und in einer zentralen *Offline-Datenbank* zusammengeführt, die auf die entsprechende Problemstellung der Simulation angepasst ist ([RB01]).

Es stellt sich nun die Frage, wie Daten aus originären Datenquellen gewonnen werden können. Hierzu existieren bereits verschiedene Ansätze, die heterogene Datenquellen mit Hilfe von Integrationsarchitekturen auf der Basis von Middleware-Datenmodellen verwenden ([H⁺99],[LRO96]). Weiterhin wurden verstärkt leistungsfähige Integrationsmechanismen zur Organisation und zum Suchen von Daten auf Basis von XML entwickelt ([D⁺98],[LRO96]).

Im Rahmen dieser Arbeit findet ein von Jensen und Hotz beschriebener Ansatz in Form einer XML-Schnittstelle Anwendung, der eine effiziente Anbindung und Zusammenführung von verteilten Informationen ermöglicht ([JH06]). Diese Schnittstelle wird auf einem Webserver als Web Service zur Verfügung gestellt und fügt sich somit in den für die Architektur verwendeten verteilten Ansatz ein.

Die lokale Verteilung der benötigten Daten auf verschiedene Systeme bedingt unterschiedliche Aufgaben zwischen einer Datenanfrage und der entsprechenden Antwort. Sind durch eine Anfrage mehrere Datenquellen betroffen, so muss diese in mehrere Unterabfragen aufgliedert und ausgeführt werden. Die Ergebnisse dieser Abfragen sind anschließend wieder zusammenzuführen.

Diese Aufgaben werden durch so genannte *Engines* übernommen. Im Allgemeinen produziert eine Engine aus einem bestimmten Input ein bestimmtes Output ([Wik07h]). Eine *Reformulation Engine* stellt die erste Stufe dar und zerlegt eine primäre Anfrage an den Webserver in Unterabfragen. Diese primäre Anfrage muss dabei diejenigen Informationen enthalten, die eine Identifikation der gewünschten Daten in den betroffenen Systemen ermöglicht. Da Ergebnisse von Unterabfragen zur Durchführung weiterer Abfragen Voraussetzung sein können, muss eine Abarbeitungsreihenfolge zusammen mit den zu übergebenden Parametern definiert sein. Daraus ergibt sich implizit auch die Reihenfolge der abzufragenden Systeme.

Die eigentliche Ausführung der Unterabfragen findet in einem weiteren Schritt, in der so genannten *Execution Engine* statt. Hier wird die eigentliche Verknüpfung der Simulationsanfrage mit realen Datenquellen in Echtzeit durchgeführt. Hierzu müssen die physikalischen Adressen der entsprechenden Datenbanken, die Instanznamen und die Beschreibung der Datenbankabfragetechnik wie *SQL (Structured Query Language)*, *Batch* oder *File-Exchange* bekannt sein. Simulationsrelevante Datenfelder sind ebenso zu definieren wie die entsprechenden Schlüsselfelder zwischen unterschiedlichen Datenbanken. Diese Konfiguration kann mit Hilfe einer komplexen XML-Struktur beschrieben werden, die anschließend als so genannter *Datasource Catalog* zur Verfügung gestellt wird.

Die Ergebnisse der Unterabfragen müssen in der Execution Engine zu einer Gesamtstruktur zusammengeführt werden. Diese Gesamtstruktur ist über die Anfragekonfiguration aus der Reformulation Engine gegeben und kann mit Hilfe von XML verwaltet werden. In den Datenbanken tauchen dabei Feldnamen auf, die mit der standardisierten Struktur von XML nicht vereinbar sind. Aus diesem Grund werden die unstrukturierten Daten aus den Datenbanken über eine Liste aus Feldnamen und den dazugehörigen Feldinhalten zur Verfügung gestellt.

In der Execution Engine wird dabei ein XML-Schema zur Verwaltung der Anfragekonfiguration von Sichten bzw. Abfragen und deren gegenseitige Verknüpfung verwendet. In der Reformulation Engine wiederum finden sich Schemata zur Verlinkung von Datenbankparametern mit simulationsrelevanten Daten.

Eine spezifische Abarbeitungsreihenfolge kann eine rekursive Abfrage über mehrere Datenbanksysteme notwendig machen. Deshalb muss es möglich sein, verschiedene Sichten rekursiv zu sinnvollen Reihenfolgen über mehrere Ebenen hinweg zu kombinieren. Dadurch können Abfragen über mehrere Tabellen oder gar Datenbanksysteme in einer Baumstruktur miteinander verknüpft und in XML dargestellt werden.

Wird nun eine primäre Anfrage an den Webserver gestellt, so ergibt sich mit Hilfe der zu übergebenden Parameter eine komplette Abfragestruktur. Daraus können in einfacher Weise SQL-Strings generiert werden. Erst die *Merge Engine* verknüpft die Werte aus den Datenbanken über deren Feldnamen mit den applikationsspezifischen Parametern, die über die Anfragekonfiguration aus der Reformulation Engine bekannt sind. Die von den Datenbanken zurückgelieferten Ergebnisse werden in einem von einer spezifischen Applikation unabhängigen XML-Format dargestellt. Abbildung 4.6 verdeutlicht diese Zusammenhänge in übersichtlicher Form.

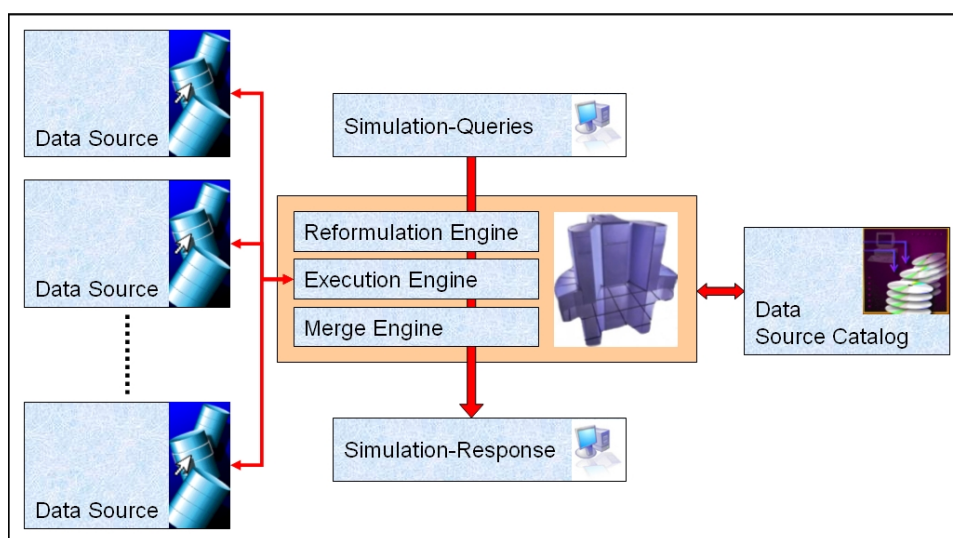


Abb. 4.6: Komponenten der allgemeinen XML-Schnittstelle ([JH06])

Diese Daten enthalten die Ergebnisse für spezifische Abfragen zur Datenversorgung eines SEWS und können über eine eindeutige *ID (Identifier)* mit dem in Abschnitt 4.1.3 vorgestellten XML-Schema verknüpft werden. Anschließend sind die vorliegenden Ergebnisse der Merge Engine mit Hilfe von *Stylesheets* in das für SEWS spezifische XML-Format zu bringen und durch XUpdate-Befehle in der XML-Datenbank abzuspeichern (vgl. Abschnitt 4.1.4).

Große Abfragen oder starke Verschachtelungen über mehrere Ebenen hinweg beeinträchtigen die Abfrageperformance. Zur Performancesteigerung können die Abfragen mit Hilfe von

Threads parallelisiert und es kann ein Connection-Pooling verwendet werden. Diese Ansätze werden an dieser Stelle nicht näher erläutert. Hierzu sei auf Jensen und Hotz verwiesen ([JH06]).

4.1.6 Aktualisierung von Modelldaten

Die Qualität der Prognosen eines SEWS ist unter anderem davon abhängig, wie aktuell die verwendeten Eingangsdaten über das reale System für die Simulation sind. Um ein Veraltern des Modells und somit die Prognosegenauigkeit nicht negativ zu beeinflussen, müssen die Modelldaten regelmäßig *gewartet* werden. Die manuelle Wartung durch einen Experten wäre zeitraubend, aufwändig und somit teuer. Mit Hilfe der in den Informationssystemen des realen Materialflusssystem hinterlegten Informationen (vgl. Abschnitt 2.2.1) über die Betriebsdaten können die Modelldaten in automatisierter Weise angeglichen werden. Die hierfür notwendigen Daten sind in dem für SEWS entwickelten XML-Schema beschrieben und es erfolgt eine Verwaltung in der XML-Datenbank (vgl. Abschnitt 4.1.3 und 4.1.4).

Während die prognoserelevanten Daten ersten Grades bei jeder zyklisch auftretenden Prognose ermittelt werden, sind für die prognoserelevanten Daten zweiten Grades geeignete Aktualisierungszeitpunkte festzulegen. Dies kann grundsätzlich auf zwei Wegen geschehen. Zum einen können so genannte *Listener* verwendet werden, die bei Veränderungen der Originaldaten tätig werden und die SEWS-internen Daten aktualisieren ([Bal01]). Zum anderen sind feste Zeitpunkte vereinbar.

Die Anwendung von Listenern ist kompliziert und zeitaufwändig, bietet aber die absolute Sicherheit, dass alle notwendigen Modelldaten jederzeit aktuell sind. Feste Zeitpunkte sind einfach realisierbar, bergen allerdings die Gefahr in sich, nicht aktuelle Modelldaten zu liefern. Dies ist allerdings von den Aktualisierungsmechanismen der zugrunde liegenden Informationssystemen abhängig. Die meisten Systeme zur Verwaltung prognoserelevanter Daten zweiten Grades werden bei Änderungen nicht sofort aktualisiert. Hierzu dienen so genannte nächtliche oder wöchentliche *Batch-Läufe*, die Veränderungen zu einem bestimmten Zeitpunkt durchführen, um die Systemlast zu optimieren ([GB03]). Aus diesem Grund genügt es, die Modelldaten ebenfalls zu einem bestimmten Zeitpunkt zu aktualisieren.

Eine vollständige Automatisierung ist unter Umständen nicht möglich. So werden Schichtmodelle nicht immer in zugänglichen Informationssystemen verwaltet. Die entsprechenden Daten müssen bei Veränderungen durch den Anwender eines SEWS also manuell angepasst werden. Dies ist sowohl von Unternehmen zu Unternehmen als auch von Materialflusssystem zu Materialflusssystem unterschiedlich und bei der Implementierung eines SEWS zu berücksichtigen.

Gelingt eine vollständige Automatisierung der Modellaktualisierung, so können die Real- und Modelldaten konsistent gehalten werden. Ansonsten spielt weiterhin die Komponente Mensch eine große Rolle, der die automatisch aktualisierten Daten final verändern kann. Dies geschieht bspw. aufgrund vorhandenem und über den Stand der Informationssysteme hinausgehendem Wissen oder Erfahrungen.

4.2 Simulation als eingebettete Applikation

Die Verwendung von Simulationsmodellen als eingebettete Anwendung in anderen Softwarewerkzeugen ist ein in der Literatur diskutiertes Themengebiet mit vielversprechenden Entwicklungspotenzialen. Bspw. bei Anwendungen in der simulationsbasierten Termin- oder Produktionsplanung ([Bru00]), zur Unterstützung von Steuer- und Regelsystemen ([Car00]) oder als Komponente in *Decision-Support-Systemen (DSS)* ([Mab00]) konnte sich die Simulation etablieren. Bereits Gibson sieht im Jahre 2000 die Einsatzmöglichkeit der Simulation als Prognosekomponente für die Unterstützung einer Ablaufplanung in Echtzeit und der Unterstützung von Kontrollfunktionen ([Gib00]).

Die in Abschnitt 3.2.2 dargestellten allgemeinen Anforderungen an SEWS haben Auswirkungen auf die verwendeten Simulatoren und Simulationsmodelle. Bestimmte Anforderungen können durch die Fähigkeiten und Funktionalitäten von Simulatoren und Simulationsmodellen nicht oder nur bedingt beeinflusst werden.

Die Komponente Simulation hat keinen Einfluss auf die Anforderung der Benutzerfreundlichkeit, da die Simulation als eingebettete Applikation verwendet wird und vor dem Anwender eines SEWS verborgen bleibt. Weiterhin ist sie eine passive Komponente zur Prognose künftiger Systemzustände. Aus diesem Grund ist auch die Anforderung der Lernfähigkeit nicht durch ein Simulationsmodell leistbar, sondern muss durch eine andere SEWS-Komponente übernommen werden.

Zu den nur eingeschränkt beeinflussbaren Anforderungen gehören die Standardisierung bzw. Wiederverwendbarkeit, das reaktive und proaktive Verhalten, die Skalierbarkeit bzw. Adaption und der Automatismus. Hierzu folgende Erläuterungen:

- **Standardisierung / Wiederverwendbarkeit**

Die Standardisierung bzw. Wiederverwendbarkeit bezieht sich auf das Gesamtsystem und seine Komponenten. Dies kann für die Simulationsmodelle nur eingeschränkt gelten, da diese speziell für ein bestimmtes reales System entwickelt werden. Im Zuge der Modellentwicklung existieren aber Module und Funktionalitäten wie bspw. Kommunikationsmechanismen, Berechnungsfunktionen und Statistiken, die zur Einbindung in ein SEWS notwendig sind. Sind diese einmal simulatorspezifisch vorhanden, können sie entsprechend weiterverwendet werden.

- **Reaktivität und Proaktivität**

Die Reaktivität und Proaktivität wird von den verwendeten Simulationsmodellen lediglich durch die Durchführung von Simulationsläufen unterstützt. Die Bewertung und Verwaltung der Ergebnisse, die Generierung von Handlungsalternativen und die Aufbereitung und Ermittlung der Realdaten erfolgt mit Hilfe unterschiedlicher Komponenten des Frameworks (vgl. Abschnitt 3.3.2). Dadurch werden die allgemeinen Anforderungen an die Modellentwicklungen zur Abbildung realer Systeme nicht berührt.

- **Skalierbarkeit / Adaption**

Bei der Modellentwicklung muss entschieden werden, welche und in welchen Wertebereichen simulationsrelevante Daten parametrierbar sein müssen. Deren Validierung,

Aktualisierung und Parametrierung wird wiederum nicht durch die Simulation, sondern durch andere Komponenten des Frameworks durchgeführt.

- **Automatismus**

Die Anforderung des Automatismus wird durch die Simulation insoweit unterstützt, indem der Simulator bzw. das Simulationsmodell bei der Durchführung von Simulationsläufen mit spezifischen Eingangsparametern stabil und fehlerfrei läuft. Dies ist zum einen durch eine geeignete Modellerstellung abzusichern und zum anderen durch eine geeignete Simulatoreauswahl zu unterstützen.

- **Unabhängigkeit**

Ein Simulator wird in einem SEWS als eingebettete Applikation und somit *fertiges Produkt* verwendet, welches in einer speziellen Programmiersprache realisiert wird. Die Ausführung dieser Softwarepakete ist abhängig von einer bestimmten Laufzeitumgebung. Aus diesem Grund ist die Anforderung der Unabhängigkeit durch die Simulation beeinflussbar. Dagegen können die Komponenten des Frameworks zur Einbindung der Simulation plattformunabhängig und verteilt gestaltet werden, wodurch das Problem leicht zu überwinden ist.

Diese Anforderungen können durch die Hauptkomponente Simulation nicht oder nur bedingt beeinflusst werden. Dagegen gehört die Performance zu denjenigen Anforderungen, die sehr stark von dem verwendeten Simulator und den Simulationsmodellen abhängig sind.

Von großer Bedeutung für die Performance eines SEWS ist die Geschwindigkeit bei der Durchführung von Simulationsläufen. Diese wird sowohl vom eingesetzten Simulator als auch durch den verwendeten *Level of Detail (LOD)* bei der Modellentwicklung beeinflusst. Der Begriff des Level of Detail ist vor allem durch den Bereich der 3D-Computergrafik geprägt, womit die verschiedenen Detailstufen bei der Darstellung virtueller Welten beschrieben wird ([Sch99]). Im Bereich der Simulation wird unter dem Detaillierungsgrad die Abstraktion und Vereinfachung im Vergleich zum realen System verstanden. Hierauf wurde bereits in Abschnitt 2.2.3.2 eingegangen.

4.2.1 Relevante Eigenschaften von Simulatoren

Die relevanten Eigenschaften von Simulatoren zur Verwendung in SEWS leiten sich aus den Anforderungen ab, die bei dem Design, der Entwicklung und dem Betrieb von Simulationsmodellen zu berücksichtigen sind (vgl. Abschnitt 3.2.2). Der Einsatz der Simulation dient vor allen Dingen zur Prognose künftiger Systemzustände.

Neben den notwendigen Anforderungen an die Modellierung müssen diejenigen Anforderungen erfüllt werden, die im Zusammenhang mit Frühwarnsystemen mit informationstechnologischem Hintergrund auftreten. Dabei handelt es sich um die Performance, die Unabhängigkeit, die Standardisierung bzw. Wiederverwendbarkeit, die Erweiterbarkeit und die Skalierbarkeit bzw. Adaptionsfähigkeit. Im Folgenden werden die sich daraus ergebenden relevanten Haupteigenschaften von Simulatoren erläutert.

- **Performance**

Die Leistungsfähigkeit eines SEWS ist vom verwendeten Simulator abhängig. Je schneller Simulationsexperimente durchgeführt werden, desto positiver wird die gesamte Performance von SEWS beeinflusst. Dabei sind die simulatorinternen Verwaltungsstrukturen zur Organisation der Simulationsmodelle und Simulationsläufe von großer Bedeutung. Vor allem die effiziente Verwaltung von Listen der aktuellen und zukünftigen Ereignisse, sowie von bedingten oder nicht bedingten Verzögerungen wirken sich sehr stark aus. Dabei ist der Aufbau und die Anzahl der Listen von Simulator zu Simulator verschieden. Ein gutes Beispiel im Performancevergleich stellt die bedingte Verzögerung mit Hilfe der `[WAIT UNTIL]`-Anweisung dar. Der Simulator *AutoMod* von dem Unternehmen *Brooks* ([Bro06]) bspw. verwendet eine globale Verzögerungsliste zur Verwaltung seiner Objekte, die auf die Erfüllung einer Bedingung warten müssen. In dem Simulator *SLX* von der Firma *Wolverine* ([Wol06]) finden sich dagegen Kontrollvariablen zur Darstellung von Elementen, die jeweils eine eigene Verzögerungsliste besitzen. Dadurch sind nur diejenigen Objekte betroffen, die auch tatsächlich über eine Kontrollvariable mit einer Bedingung verknüpft sind. Deshalb ist die Verwendung der Anweisung `[WAIT UNTIL]` zur bedingten Verzögerung in *SLX* wesentlich effizienter ([SB06]). Ein detaillierter Performancevergleich bezüglich bedingter Verzögerungen zwischen *AutoMod* und *SLX* findet sich im Anhang B.

Weiterhin wirkt sich die Art der Ausführung des Quellcodes der Simulationsmodelle aus. Zum einen besteht die Möglichkeit, den Quellcode zunächst zu analysieren, zu kompilieren und anschließend Simulationsläufe durchzuführen. Zum anderen kann ein Interpreter zum Einsatz kommen, der den Quellcode zur Laufzeit des Programmes einliest, analysiert und ausführt. Der allgemeine Nachteil beim Einsatz von Interpretern ist die im Vergleich zu kompilierten Programmen deutlich langsamere Ausführungsgeschwindigkeit. Dies gilt auch für die im Bereich von Produktion und Logistik eingesetzten Simulatoren. Beispiele für Materialflusssimulatoren, die eine Interpretertechnologie verwenden, sind *eM-Plant* von der Firma *UGS* und *Quest* von der Firma *Delmia*.

- **Schnittstellen und Integrationsmöglichkeiten**

Die heute am Markt gängigen Simulatoren besitzen eine offene Systemarchitektur, die vielfältige Schnittstellen und Integrationsmöglichkeiten bieten. Dies reicht von Schnittstellen zur Datenintegration wie *ActiveX*, *CAD*, *COM*, *DDE*, *ODBC*, *SQL*, *SDX* oder *XML* bis hin zur Visualisierungsunterstützung mit *HTML*, *WMV*, *AVI* oder andere Audio- und Videoformate ([SH98],[Wol06],[Bro06]). Die Verwendung dieser Technologien zur Integration der Hauptkomponente Simulation in ein SEWS muss dabei den Anforderungen Unabhängigkeit, Automatismus, Standardisierung bzw. Wiederverwendbarkeit und wiederum Performance genügen. Da vielfältige Kommunikationsmechanismen existieren und die SEWS-interne Datenhaltung mit Hilfe von XML geschieht, bietet sich die Verwendung von Stylesheets zur Transformation von XML-Daten in ein simulatorspezifisches Format an ([Lov02]). Hierdurch ist eine Entkopplung von einer bestimmten Technologie möglich. Auf diese Entkopplung wird in den Abschnitten 4.3.1.1 und 4.3.2.1 eingegangen.

- **Konfigurierbarkeit und Erweiterbarkeit**

Die Anforderung der Konfigurierbarkeit bezieht sich auf eine einfache Veränderung

von Simulationsparametern in Form simulationsrelevanter Daten. Geeignete Simulatoren müssen eine weitestgehende und einfache Parametrierung von Modellen ermöglichen, da eine Veränderung des Simulationsmodells zu viel Zeit in Anspruch nehmen würde und kaum zu automatisieren wäre. Dabei muss ein Abgleich mit dem in SEWS verwendeten Datenmodell erfolgen (vgl. Abschnitt 4.1.3). Alle dort verwendeten Informationen sind veränderbar und müssen somit jederzeit im Simulationsmodell aktualisierbar sein.

Die Anforderungen der Erweiterbarkeit, Skalierbarkeit und Adaptionfähigkeit sind hierbei ebenfalls von großer Bedeutung. Hier ist ein einfaches Mapping der Informationen im Datenmodell des SEWS-Frameworks auf das Datenmodell des Simulators erforderlich. Dadurch wird die konsistente Darstellung der Zustandsinformationen des realen Systems sowohl im SEWS-Framework als auch im Simulationsmodell ermöglicht.

Simulatoren bieten zur Verwaltung eines Datenmodells unterschiedliche Ansätze wie beispielsweise Listen, Felder, Variablen und Tabellen. Die Ansätze sind hierbei von Simulator zu Simulator verschieden und für den Simulationsexperten bei der Modellierung einfacher oder schwieriger zu verwenden. Vernachlässigt man diesen unterschiedlichen Aufwand, reduziert sich die Anforderung an einen Simulator auf die generelle Abbildungsfähigkeit des Datenmodells in SEWS.

- **Weitere Funktionalitäten**

Der Simulator muss die freie Programmierung verschiedener Statistik-, Berechnungs- und Ausgabefunktionen unterstützen. Dadurch wird die Integration eines Simulationsmodells in SEWS erst ermöglicht, da die hierfür notwendigen Komponenten und Funktionen standardmäßig nicht vorhanden sind. Weiterhin muss die Möglichkeit existieren, mehrere Instanzen eines Simulationsmodells zu erzeugen und parallel zu verarbeiten. Findet die parallele Ausführung auf nur einem Rechner statt, so wirkt sich das nachteilig auf die Performance aus. Deshalb ist eine Verteilung der Simulationsläufe auf mehrere Rechner notwendig.

- **Lizenzkosten**

Die Art der Lizenzmodelle ist ebenfalls von Bedeutung. Lizenzen für Materialflusssimulatoren im Bereich von Produktion und Logistik zeichnen sich durch hohe Lizenzkosten aus, die sich im fünfstelligen Euro-Bereich bewegen. Dies ist ebenfalls ein begrenzender Faktor bei der Parallelisierung, da gleichzeitig mehrere Instanzen des Simulationsmodells verwendet werden müssen.

- **Ansteuerbarkeit**

Unter der Ansteuerbarkeit soll die Möglichkeit verstanden werden, einen Simulator aus anderen Applikationen oder einen Programmcode heraus zu verwenden. Dies ist besonders bei der Entwicklung eines *Web Service Simulator* von Bedeutung, um ein Simulationsmodell aufzurufen und Simulationsexperimente durchführen zu können. Sowohl mit SLX als auch mit AutoMod entwickelte Simulationsmodelle können über einen einfachen Kommandozeilenbefehl gestartet werden. Dies erleichtert die Kapselung und Bereitstellung der Simulation als Dienst in einer serviceorientierten Architektur ([Bie06]).

Die Komplexität und der Aufwand des Modellaufbaus spielt für die Performance während des

Betriebes eines SEWS eine untergeordnete Rolle. Es ist nur darauf zu achten, dass bei der Modellierung alle Möglichkeiten zur Performancesteigerung ergriffen werden. Weiterhin sind die im folgenden Abschnitt dargestellten Anforderungen an Simulationsmodelle zu unterstützen.

4.2.2 Anforderungen an Simulationsmodelle

Die Anforderungen an Simulationsmodelle ergeben sich aus den notwendigen Funktionalitäten und Komponenten zur Integration in ein SEWS. Dabei ist deren Realisierung und Umsetzung abhängig vom verwendeten Simulator, der durch einen Web Service gekapselt wird. Dieser übernimmt die Daten vom SEWS-Framework, versorgt damit die Simulation, steuert den Simulator und stellt die Ergebnisse der Simulationsläufe bereit. Hierzu sind im Simulationsmodell folgende Funktionalitäten notwendig:

- **Data Import**

Es muss eine Komponente im Simulator bzw. Simulationsmodell existieren, welche die vom Web Service in einem simulatorspezifischen Datenformat bereit gestellten Daten importieren kann.

- **Model Initialization**

Eine weitere erforderliche Funktionalität initialisiert das Simulationsmodell mit dem gegenwärtigen Zustand des realen Systems. Weiterhin wird das Modell mit den notwendigen Informationen für den Prognosezeitraum wie bspw. die Belegungsplanung versorgt.

- **Run Control**

Nach der Initialisierung des Simulationsmodells wird ein Simulationslauf gestartet.

- **Variable Calculation**

Zu definierten Zeitpunkten oder kontinuierlich während des Simulationslaufes werden definierte Kennwerte berechnet. Diese dienen zum einen der Detektion von Ausnahmefällen und zum anderen werden diese zur Darstellung der Ergebnisse der Simulationsläufe verwendet. Die Kennwertberechnung wird in Abschnitt 4.3.2.1 näher beschrieben.

- **Exception Interpreter**

Die berechneten Kennwerte werden hinsichtlich definierter Grenzwerte untersucht. Wird einer oder mehrere Grenzwerte über- oder unterschritten, so handelt es sich um detektierte Ausnahmefälle. Wie im Abschnitt 3.4.4 bereits erläutert wurde, würde eine zentraler Exception Interpreter im SEWS-Framework zu Performanceverlusten führen.

- **Data Export**

Nach Beendigung des Simulationslaufes werden die Prognosedaten in Form der berechneten Kennwerte und detektierten Ausnahmefälle exportiert und dem Web Service zur Übergabe an das SEWS-Framework zur Verfügung gestellt.

Neben diesen Funktionalitäten muss das Prozesssystem des realen Systems in dem Simulationsmodell abgebildet werden. Dabei sind bei der Modellerstellung bestimmte Gesichtspunkte

zu berücksichtigen, um den Anforderungen bei der Implementierung von SEWS nachzukommen. Abbildung 4.7 zeigt den Ablauf und die Funktionen zur Integration der Simulation in SEWS.

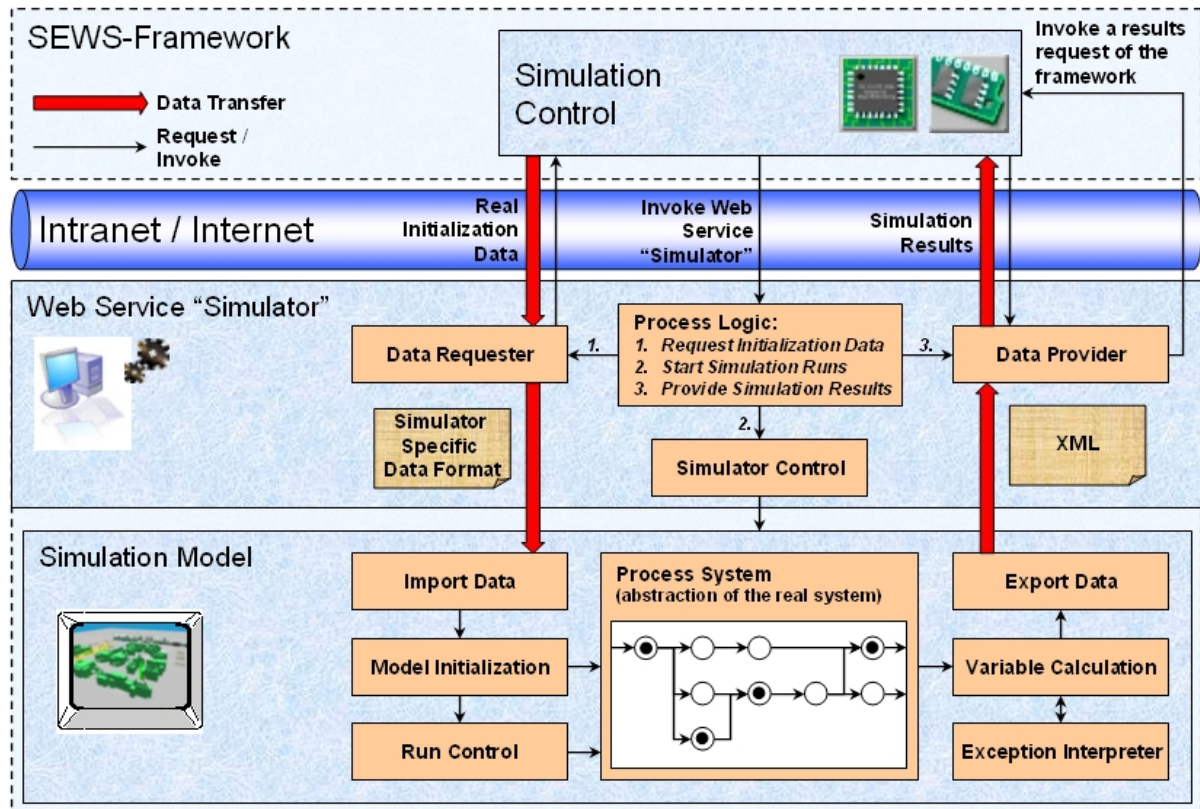


Abb. 4.7: Ablauf und Funktionen zur Integration der Simulation in SEWS

Die Prozessabfolgen, Alternativprozesse und benötigten Ressourcen der einzelnen Produkte müssen flexibel darstell- und veränderbar sein. Dies kann bspw. über Ressourcen- und Prozesslisten erfolgen, aus denen die entsprechenden Ressourcen und Prozesse über eine eindeutige *ID* ansprechbar sind. Die Reihenfolge dieser IDs wird für einen spezifischen Produkttyp in seiner Prozess-Sequenz festgelegt (vgl. Abschnitt 4.1.3).

Weiterhin sind die Messpunkte eines realen Systems abzubilden, wobei ein Messpunkt mindestens einen Vor- oder Nachfolgerprozess besitzen muss. Hierdurch wird der Status eines Produktes und der weitere Verlauf in seinem Produktionsfortschritt während der Simulation festgelegt. Dies ermöglicht die Initialisierung mit dem realen Systemzustand, da ein Produkt zum entsprechenden Zeitstempel des aktuellen Messpunktes dem nachfolgenden Prozess zur Abarbeitung übergeben wird.

Das Simulationsmodell muss beliebige statistische Verteilungen für die Prozesszeiten verarbeiten können. Weiterhin werden die wichtigsten statistischen Informationen über das Ausfallverhalten von Ressourcen durch die statistischen Verteilungen *Mean Time Between Failure (MTBF)* und *Mean Time To Repair (MTTR)* beschrieben ([AF05],[A⁺04]). Dabei ist mit Hilfe

der MTBF eine Aussage über die Wahrscheinlichkeit des Zeitraums zwischen zwei auftretenden Ausfällen von Ressourcen möglich, während die MTTR die Wahrscheinlichkeit der zeitlichen Dauer eines Ressourcenausfalls nach dessen Eintreten beschreibt.

Die Berücksichtigung von Ressourcenausfällen bei der Prognose künftiger Zustände des realen Systems erfolgt im Rahmen dieser Arbeit nicht. Die Werte der MTBF liegen im Wochen- oder gar Monatsbereich und somit im Allgemeinen weit über den Prognosezeiträumen von SEWS im Bereich von Produktion und Logistik. Dabei handelt es sich um Zeiträume von Stunden, Schichten oder Tagen. Da aus diesem Grund die Prognose von Ressourcenausfällen keine Rolle spielt, soll sie hier vernachlässigt werden.

Die für das reale System gültigen und einsetzbaren Schichtmodelle sind ebenfalls zu berücksichtigen. Diese sind über eine eindeutige ID abrufbar zu gestalten und es ist sicherzustellen, dass alle von einem Schichtmodell betroffenen Ressourcen damit umgehen können.

Ein komplexes Problemfeld ist die Darstellung unterschiedlicher Steuerungskonzepte für den Materialfluss. So kann beispielsweise *Just-In-Sequence*, *Kanban* oder *ConWIP* angewendet werden ([AF05],[A⁺04]). Diese Steuerungsprinzipien gelten für das gesamte zu betrachtende Materialflusssystem oder nur in bestimmten Teilbereichen. Dabei besteht eine sehr starke Abhängigkeit vom spezifischen Materialflusssystem, welche einzelnen Prozesse bei einer Veränderung betroffen sind. Solche Prozesse können zur Sortierung, Priorisierung oder Sammlung (z.B. *Batch-Prozesse*) dienen. Aus diesem Grund müssen bei der Entwicklung eines Simulationsmodells für SEWS alle relevanten Prozesse im realen Materialflusssystem erfasst und ihre Eigenschaften in Abhängigkeit vom Steuerungskonzept analysiert werden.

4.3 Simulation zur Vorhersage künftiger Zustände des realen Systems

Um die Simulation als eingebettete Applikation verwenden zu können, muss diese verschiedene Funktionen übernehmen bzw. unterstützen. Dazu gehören die Prognose künftiger Zustände des realen Systems, die Detektion von Ausnahmefällen und die Bewertung generierter Handlungsalternativen. In diesem Abschnitt wird näher auf die Prognose und die Detektion von Ausnahmefällen eingegangen, wobei die hier erläuterten grundlegenden Funktionalitäten auch bei der späteren Generierung von Handlungsalternativen Anwendung finden.

Die Verwendung der Simulation als Prognosetool erfordert die Betrachtung mehrerer Themenschwerpunkte. Hierzu gehört die Initialisierung von Simulationsmodellen, die eigentliche Prognose künftiger Systemzustände, die Detektion von Ausnahmefällen, die Modellaktualisierung und die Problematik der Prognosegenauigkeit.

4.3.1 Initialisierung von Simulationsmodellen

Die Initialisierung von Simulationsmodellen mit dem gegenwärtigen Zustand des realen Systems ist ein notwendiger Vorgang, um Prognosen über künftige Systemzustände treffen zu

können (vgl. Abschnitt 3.3.1). Das Ziel liegt dabei in einer realitätsnahen Abbildung des realen Systemzustandes zu Beginn des Prognosezeitraumes. Unter der Initialisierung wird in der Informatik im Allgemeinen

„[...] der Teil des Ladevorganges eines Programms, in dem der zur Ausführung benötigte Speicherplatz (z.B. Variablen, Code, Buffer, ...) für das Programm reserviert und mit Startwerten gefüllt wird [...]“ ([GB03])

verstanden. Bezieht man dies auf die Initialisierung von Simulationsmodellen mit dem gegenwärtigen Systemzustand eines realen Systems zu Prognosezwecken, so ergibt sich folgende Definition:

„Die Initialisierung von Simulationsmodellen entspricht dem Start eines Simulationsmodells und der Belegung der modellinternen Kontrollstrukturen mit Startwerten in der Art und Weise, dass nach erfolgter Initialisierung das Simulationsmodell den gegenwärtigen Zustand eines realen Systems mit hinreichender Genauigkeit zu Prognosezwecken repräsentiert.“

Die Initialisierung gliedert sich dabei in zwei grundsätzliche Abschnitte, die im Folgenden näher erläutert werden. Es handelt sich dabei um die Datenübertragung bzw. -bereitstellung und um den eigentlichen Initialisierungsvorgang.

4.3.1.1 Datenübertragung und -bereitstellung

Bevor das Simulationsmodell initialisiert werden kann, müssen dem Simulator bzw. Simulationsmodell SEWS- und simulationsrelevante Daten zur Verfügung gestellt werden (vgl. Abschnitt 4.1.1). Je nach verwendetem Simulator sind diese Daten in ein simulatorspezifisches Format zu überführen, das dieser verarbeiten kann. Für diesen Transformationsprozess wird *XSLT (XSL Transformation)* verwendet ([Lov02]).

XSLT ist Teil der *Extensible Stylesheet Language (XSL)* und ist eine Programmiersprache zur Transformation von XML-Dokumenten. Hierzu werden so genannte *XSLT-Stylesheets* verwendet, die nach den Regeln des XML-Standards aufgebaut sind. Mit Hilfe eines *XSLT-Prozessors* wird ein *XSLT-Stylesheet* eingelesen und ein XML-Dokument in ein gewünschtes Ausgabeformat wie beispielsweise *XHTML*, beliebige *TXT*-Formate oder in ein anderes XML-Dokument transformiert ([Lov02]).

Dadurch werden die Anforderungen der Wiederverwendbarkeit, Standardisierung und Benutzerfreundlichkeit konsequent unterstützt. Die Datenverwaltung und -bereitstellung wird von einer spezifischen Simulatortechnologie entkoppelt, da erst mit Hilfe eines *XSLT-Stylesheets* simulatorspezifische Formate erzeugt werden. Diese Entkopplung kommt auch bei der im nachfolgenden Abschnitt 4.3.2 näher erläuterten Prognose und Detektion von Ausnahmefällen zum Tragen.

Die Transformation erfolgt durch den Web Service *Simulator* unter Verwendung eines simulatorspezifischen *XSLT-Stylesheets*. Das Ergebnis der Transformation wird auf dem Rechner

lokal abgelegt, auf dem die Simulation durchgeführt wird, und steht anschließend dem Simulator bzw. Simulationsmodell zur eigentlichen Initialisierung zur Verfügung.

4.3.1.2 Initialisierung

Wie zu Beginn dieses Abschnittes erläutert wurde, ist unter der Initialisierung von Simulationsmodellen das Mapping des realen Systems auf das Simulationsmodell unter der Verwendung geeigneter Beschreibungsdaten des gegenwärtigen Systemzustands zu verstehen. Der Vorgang der Initialisierung beginnt dabei mit der Übertragung der initialisierungsrelevanten Daten, die anschließend vom Simulationsmodell einzulesen oder zu importieren sind. Diese stehen dann dem eigentlichen Initialisierungsvorgang zur Verfügung. Dabei handelt es sich um:

1. **Initialisierung notwendiger Funktionskomponenten**

Bei den notwendigen Funktionskomponenten handelt es sich um die Kennwertberechnung und Ausnahmefalldetektion, deren Kontrollstrukturen vor dem Beginn der Simulationsläufe zur Verfügung stehen müssen.

2. **Instantiierung der Simulationsobjekte**

Die im Simulationsmodell benötigten Simulationsobjekte wie Ressourcen, Werker, Produkte oder Variablen werden instantiiert.

3. **Anpassung der Stati der Simulationsobjekte**

Zur Darstellung des gegenwärtigen Zustand des realen Systems müssen die zu diesem Zeitpunkt geltenden Stati der instantiierten Simulationsobjekte angepasst werden. Dies bezieht sich bspw. auf den Status der Produkte im Bearbeitungsprozess, der Kalkulation von Restbearbeitungszeiten oder auf den Status von Produkten.

4. **Synchronisation der Simulationsuhr**

Bevor die Simulationsläufe zu Prognosezwecken für einen bestimmten Prognosezeitraum gestartet werden können, ist die Simulationsuhr mit dem Zeitpunkt des gegenwärtigen Systemzustandes zu synchronisieren.

Zur Initialisierung derjenigen dynamischen Simulationsobjekte im Simulationsmodell, die Produkte in einem Materialflusssystem repräsentieren, kommen drei grundsätzliche Initialisierungsstrategien in Frage, die im Folgenden diskutiert werden.

Strategie 1

Hierzu wird lediglich der Eintrittszeitpunkt t_i jedes Produktes i in das Materialflusssystem benötigt. Zu diesen Zeitpunkten sind die Simulationsobjekte zu erzeugen, welche die Produkte repräsentieren. Anschließend werden diese in das Materialflusssystem eingeschleust und durchlaufen dieses gemäß ihrer Prozess-Sequenz. Der Zeitstempel des ältesten Produktes bildet dabei einen zeitlichen Bezugspunkt α in der Vergangenheit, bei dem der Simulationslauf beginnt. Relativ zu diesem Bezugspunkt wird jedes Produkt i mit der Zeitdifferenz Δt_i in das Modell eingeschleust, wobei bei n Produkten

$$\Delta t_i = \alpha - t_i, i \in [1, \dots, n]$$

gilt. Das Modell ist anschließend synchronisiert und die Prognose beginnt, wenn jedes Produkt eingeschleust wurde und die Simulationsuhr mit dem Initialisierungszeitpunkt übereinstimmt. Diese Strategie findet dann Anwendung, wenn ungenaue Messungen an den Messpunkten vorliegen oder keine bzw. eine unzureichende Anzahl von Messpunkten vorhanden ist.

Strategie 2

Eine weitere Strategie verwendet die Reihenfolge der Produkte an den einzelnen Messpunkten. Dabei wird jedes Produkt i einer Liste seines letzten Messpunktes zugeordnet, die nach deren Zeitstempel t_i aufsteigend sortiert wird. In dieser Reihenfolge werden die Produkte an den entsprechenden Messpunkten in das Materialflusssystem eingeschleust, die anschließend dem weiteren Verlauf ihrer Prozess-Sequenzen folgen. Ist das letzte Produkt eingeschleust und die Simulationsuhr mit dem Initialisierungszeitpunkt synchron, beginnt die Prognose. Diese Strategie findet Anwendung, falls ein Simulator keine Möglichkeit zur Zeitverzögerung von Simulationsobjekten um ein bestimmtes Δt_i bieten sollte.

Strategie 3

Hier wird die Strategie 2 um die Zeitverzögerung von Produkten erweitert. Dabei wird wiederum ein $\Delta t_{i,j}$ relativ zu einem Bezugspunkt α berechnet. Dieser Bezugspunkt stellt allerdings diesmal den Zeitstempel des ältesten Zeitstempels eines Produktes der Produktlisten über alle Messpunkte hinweg dar. Es existieren also bei m Messpunkten für jeden Messpunkt j Produkte mit den Bezugspunkten α_j . Daraus ergibt sich $\alpha = \min \alpha_j \forall j$, mit $j = 1, \dots, m$, als globaler Bezugspunkt, zu dem wiederum für jedes Produkt ein spezifisches Δt_i berechnet werden kann, zu dem das Produkt an seinem Messpunkt eingeschleust wird. Ist das letzte Produkt zu seinem Zeitstempel eingeschleust worden, muss wiederum die Simulationsuhr synchronisiert werden. Anschließend ist die Initialisierung abgeschlossen und der Prognosezeitraum beginnt.

Bei einem abschließenden Vergleich der Strategien ist leicht einzusehen, dass Strategie 3 einen kürzeren Synchronisationszeitraum benötigt als Strategie 1. Weiterhin erzielt diese Strategie die präziseste Initialisierung, die anhand vorhandener Messpunkte in einem Materialflusssystem erreichbar ist. Allerdings stellt diese auch die höchsten Anforderungen an die Datenqualität, Datenaktualität und Datenerfassung.

Es existieren unterschiedliche Ansätze zur Initialisierung im Bereich der Online-Simulation. Typische Anwendungsgebiete sind sowohl proaktive *Decision-Support-Systeme* für Scheduling-Probleme in Produktionssystemen ([GSS02],[CSG02],[GW02]) als auch Verkehrssimulationen und die Simulation von Personenströmen ([MCH⁺04],[HTRS03]). Hanisch et al. diskutiert explizit die Initialisierung bei der Online-Simulation zu Prognosezwecken ([HTS05]). Dabei werden zwei grundsätzliche Varianten zur Initialisierung genannt:

1. Modellgenerierung unter Verwendung von Initialisierungsdaten über den realen Systemzustand.
2. Ständige Synchronisierung eines Vatermodells mit dem realen System und Erzeugung und Initialisierung von Kindmodellen.

Die zweite Variante erfordert einen großen Steuerungsaufwand außerhalb des Simulationsmodells, da ein ständiger Abgleich des Modellzustandes mit dem realen System erfolgen muss.

Weiterhin ist der Aufbau des Simulationsmodells wesentlich komplexer, da zu jeder Zeit der Status aller Simulationsobjekte veränderbar sein muss.

Die erste Variante ist wesentlich einfacher zu implementieren. Es findet keine Modellgenerierung statt, sondern ein „leeres“ Modell wird initialisiert und repräsentiert anschließend den Zustand des realen Systems. Die Unterscheidung in dynamische und statische Informationen (vgl. Abschnitt 4.1.1) ermöglicht eine unterschiedliche Handhabung bei der Initialisierung. Die statischen Informationen werden durch die Veränderung der Stati der betreffenden Simulationsobjekte zum Zeitpunkt des Prognosebeginns berücksichtigt. Dagegen können die dynamischen Informationen das Prozess-System mit den dazwischen befindlichen Messpunkten regelrecht als „Türen“ zum Materialflusssystem benutzen. Dadurch entfällt eine komplexe Ansteuerung und das Simulationsmodell ist zum Zeitpunkt des Prognosebeginns mit einer ebenso hohen Qualität bei gleicher Ausgangssituation initialisiert wie bei der zweiten Variante.

4.3.2 Prognose von Ausnahmefällen

Eine Anforderung an SEWS ist das proaktive Verhalten, das die Prognose künftiger Zustände eines realen Systems und deren Analyse hinsichtlich möglicher Ausnahmefälle einschließt. In Abschnitt 3.2.1 wird ein Ausnahmefall als Abweichung von einer definierten oder gegebenen Regel und in diesem Zuge ein Ausnahmefall als das Auftreten negativer Effekte oder positiver Potenziale definiert. Dabei über- oder unterschreiten dedizierte Kennwerte einen oder mehrere Grenzwerte bzw. Grenzwertbereich.

Es stellt sich nun die Frage, wie solche Ausnahmefälle prognostizierbar sind. Im Folgenden wird zunächst auf die Beschreibung von Ausnahmefällen in SEWS eingegangen. Anschließend erfolgt eine Erläuterung, wie Ausnahmefälle während der Durchführung von Simulationenläufen detektiert werden können.

4.3.2.1 Beschreibung von Ausnahmefällen

Eine Anforderung an SEWS ist die Unabhängigkeit von einem spezifischen Simulator (vgl. Abschnitt 3.2.2). So vielfältig die auf dem Markt erhältlichen diskreten Materialflusssimulatoren sind, so unterschiedlich sind auch die Möglichkeiten der Kennwertberechnung, Report- und Ergebnisgenerierung während und nach Simulationenläufen.

Um die Unabhängigkeit von SEWS zu unterstützen, muss die Beschreibung von Kennwerten sowohl zum Monitoring des Realsystems als auch zur Unterstützung der Prognosefunktionalität standardisiert werden. Hierfür wird in dem globalen XML-Schema von SEWS das Kind-Element <Rules> zur Verfügung gestellt (vgl. Abschnitt 4.1.3), welches zur Definition und Parametrierung von Kennwerten eingesetzt wird (vgl. Abb. 4.8).

Diese Kennwerte sind durch den Anwender eines SEWS in Form von Berechnungsregeln frei konfigurierbar. Der Simulator bzw. das Simulationsmodell muss diese Berechnungsregeln verarbeiten können und die entsprechenden Kennwerte kontinuierlich, zu definierten Zeitpunkten

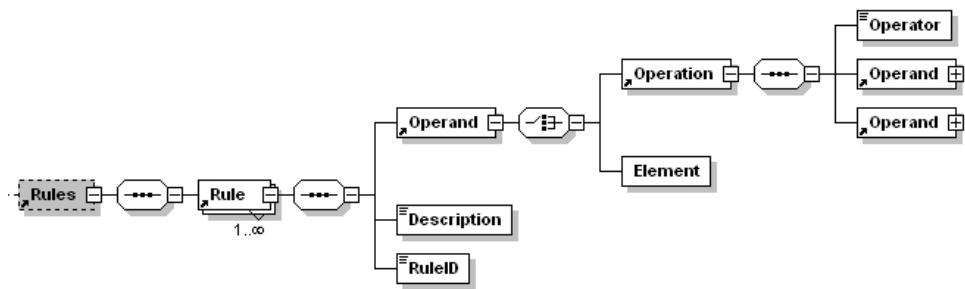


Abb. 4.8: XML Schema Definition zur Kennwertberechnung

oder nach dem Simulationslauf in einem XML-Format zurückliefern. Dabei sind sowohl kardinale, ordinale oder nominale Werte als Ergebnis von Kennwerten darstellbar. Ein Beispiel für nominale Werte wäre ein gewünschter Status einer Maschine, der sich im Prognosezeitraum nicht verändern darf.

Ein Kennwert wird hierbei durch die rekursive Verwendung des XML-Elementes `<operand>` definiert, welches entweder ein Element oder eine Operation repräsentiert. Eine Operation besteht aus einem Operator und zwei Operanden, die wiederum eine Operation oder ein Element beinhalten können. Ein Element enthält folgende Informationen zu einer korrespondierenden Simulations-Entität im Simulationsmodell:

- **ElementType**
Definiert die Art der Simulations-Entität, deren Wert in die Berechnungsregel mit einbezogen wird.
- **ElementID**
Repräsentiert die eindeutige ID einer spezifischen Simulations-Entität.
- **AttributeID**
Spezifiziert das gewünschte Attribut der Entität, die den gewünschten Wert für die Berechnungsregel repräsentiert.

Entitäten sind im Allgemeinen Repräsentationen realer Objekte ([Tha00]). Eine Berechnungsregel wird in Form eines gerichteten Baumes dargestellt ([NM02]), wobei die Blätter dieses Baumes den Attributwerten der Simulations-Entitäten entsprechen. Dieser rekursive Aufbau und die Baumstruktur der Berechnungsregeln ermöglichen eine einfache Implementierung von Programmcode zur Kennwertberechnung in den Simulationsmodellen. Die Elemente definieren die notwendigen Attribute der Simulations-Entitäten, deren Werte ermittelt und zur Berechnung herangezogen werden. Zur Verdeutlichung wird in Anhang C ein einfaches Beispiel angeführt.

Die Definition von Kennwerten in dieser Form ist nicht vollkommen neu. Bereits die *Mathematical Markup Language (MathML)* stellt eine ähnliche Systematik zur Darstellung mathematischer Ausdrücke, Symbole und Formeln zur Verfügung, um diese in *HTML*-Dokumente zu integrieren ([W3C07a]). Dabei beinhaltet sie sowohl die Präsentation als auch die Information über die Bedeutung von Formelkomponenten.

Die Aufgaben der Berechnungsregeln für Kennwerte bestehen aus der Bereitstellung der Simulationsergebnisse zum Monitoring der prognostizierten und gegenwärtigen Systemzustände und der Unterstützung der Detektion von Ausnahmefällen. Zur Definition eines bestimmten Typs von Ausnahmefällen wird ein weiteres Element im globalen XML-Schema für SEWS verwendet, das Element `<Exceptions>` (vgl. Abschnitt 4.1.3). Dieses komplexe Element beinhaltet sowohl *User Exceptions* als auch *System Exceptions*. Dabei entspricht eine *System Exception* der Verknüpfung einer Berechnungsregel mit einem oder mehreren Grenzwerten, bei deren Über- oder Unterschreitung ein Ausnahmefall vorliegt.

Die so definierten Typen von Ausnahmefälle und Berechnungsregeln sind dem Simulationsmodell zur Verfügung zu stellen, welches diese bei der Initialisierung notwendiger Funktionskomponenten berücksichtigen muss (vgl. Abschnitt 4.3.1). Hierbei ist die Implementierung der Berechnungsregeln im Simulationsmodell abhängig von dem verwendeten Simulator. Die übertragenen und mit Hilfe von XSLT in ein simulatorspezifisches Format übertragenen XML-Daten beinhalten die Definitionen der Berechnungsregeln und Ausnahmefälle. Diese Daten werden anschließend importiert bzw. eingelesen, die entsprechenden Funktionskomponenten instanziiert, die Baumstruktur der Berechnungsregeln abgebildet und mit den Ausnahmefällen verknüpft. Anschließend stehen dem Simulationsmodell Funktionen zur Berechnung von Kennwerten und der Überprüfung hinsichtlich möglicher Ausnahmefälle für die Simulationsläufe zur Verfügung. Auf die Detektion von Ausnahmefällen während der Simulationsläufe wird im Folgenden näher eingegangen.

4.3.2.2 Detektion von Ausnahmefällen

Nachdem die Initialisierung des Simulationsmodells abgeschlossen worden ist, repräsentiert dieses den Zustand des realen Systems zum Zeitpunkt der Datenerhebung bzw. zu Beginn des Prognosezeitraumes. Auf Basis dieser Ausgangssituation werden nun zu unterschiedlichen Zeitpunkten Ausnahmefälle detektiert.

Die Anforderung Reaktivität von SEWS begründet die Bereitstellung der Funktionalitäten eines Monitoringsystems (vgl. Abschnitt 3.2.2). Aus diesem Grund muss der gegenwärtige Systemzustand hinsichtlich möglicher Ausnahmefälle untersucht werden. Dies geschieht nach der Initialisierungsphase des Modells, noch vor der Erzeugung von Prognosedaten. Der Vorteil liegt darin, dass keine weiteren Funktionalitäten oder Module zur Erfüllung der Reaktivität notwendig sind, die nicht auch im Zusammenhang mit der Proaktivität nötig wären. Die Detektion von Ausnahmefällen kann auf zwei grundsätzliche Arten geschehen:

1. **Post-Run Detection**

Die Ergebnisse der Simulationsläufe werden hinsichtlich möglicher Ausnahmefälle nach deren Beendigung, also *post-run*, analysiert. Diese Vorgehensweise besitzt allerdings einige Schwachstellen. Um genauere Aussagen über eine Ausnahmesituation treffen zu können, wie bspw. wann und in welcher Höhe eine Grenzwertüberschreitung stattgefunden hat, müssten die betreffenden Simulations-Entitäten mit Hilfe einer *Trace*-Funktion kontinuierlich während des Simulationslaufes verfolgt werden. Bereits bei kurzen Simulationsläufen entstehen hierdurch enorme Datenmengen, was die Performance stark

einschränkt ([GLJ⁺01]). Weiterhin entsteht dadurch ein Mehraufwand bei der Entwicklung von SEWS. Zum einen muss eine zusätzliche Komponente existieren, die eine nachträgliche Analyse der Daten durchführt, zum anderen muss das simulatorspezifische Output-Format der Trace-Files in ein allgemeines Format überführt werden. Einige Simulatoren wie AutoMod bieten auch die Möglichkeit, Histogramm-Auswertungen zu generieren ([Bro06]), doch auch dies ist bei der expliziten Ausnahmefalldetektion nicht ausreichend.

2. Concurrent Detection

Eine weitere Möglichkeit besteht darin, während der Simulationsläufe entweder zu jedem Ereigniszeitpunkt oder nach festen Zeitintervallen eine Kontrolle der mit Ausnahmefällen verbundenen Kennwerte durchzuführen. Dabei stellt die Bestimmung der Zeitintervalle zwischen zwei Detektionszeitpunkten von Ausnahmefällen bei der Prognose künftiger Systemzustände eine Schwierigkeit dar, die in der Literatur noch nicht problematisiert worden ist. Grundsätzlich kann jedes Ereignis während des Simulationslaufes eine Veränderung relevanter Kennwerte herbeiführen und somit Ausnahmefälle verursachen.

Die zweite Variante wird favorisiert, da durch die Einführung geschickter Verfahren zur *Concurrent Detection* eine bessere Performance erreicht werden kann als dies bei Einsatz des Tracings der Fall ist.

Es stellt sich nun die Frage, wie die Zeitpunkte zur Ausnahmefalldetektion gewählt werden sollten. Da es sich im Bereich der Simulation in Produktion und Logistik bei den eingesetzten Simulatoren um ereignisgesteuerte Diskrete Simulatoren handelt (vgl. Abschnitt 2.1.2.1), besteht eine Möglichkeit in der Ausnahmefalldetektion nach jeder Ereignisbearbeitung während des Simulationslaufes. Dadurch ist keine Zeitpunktbestimmung erforderlich, doch leidet die Performance der Simulationsläufe. Hier spielt die Effizienz der realisierten Veränderungsverfolgung der entsprechenden beteiligten Variablen und Attribute im Simulator eine wesentliche Rolle (vgl. Abschnitt 4.2.1).

Eine weitere und sinnvollere Möglichkeit besteht in der Festlegung von Detektionszeitpunkten, da dies die Anforderung der Performance an SEWS unterstützt. An diesen Detektionszeitpunkten werden alle Kennwerte berechnet, die mit definierten Ausnahmefällen verknüpft sind. Dabei kann die Tatsache ausgenutzt werden, dass die Möglichkeit eines Ausnahmefalles sich nur bei einigen wenigen Zustandsänderungen in Materialflusssystemen ergeben können, die für jedes System spezifisch zu analysieren sind. Es kann sich um folgende Arten von Zustandsänderungen handeln, die im Folgenden als Z^* bezeichnet werden:

- Beginn oder Beendigung von Bearbeitungsprozessen.
- Beginn oder Beendigung von Transportvorgängen.
- Zusammenführungen oder Verzweigungen von Materialflüssen ([AF05]).
- Zugang oder Abgang von Material in Lagern, Puffern, Stellflächen o.ä.
- Beginn oder Beendigung von Schichtpausen.

Es können zwei verschiedene Varianten zur Festlegung von Detektionszeitpunkten unterschieden werden. Die erste Variante verknüpft die mit einer Zustandsänderung verbundenen Prozesse im Simulationsmodell mit einem Funktionsaufruf. Wird also beispielsweise ein Bearbeitungsprozess auf einer Maschine beendet, so wird eine Funktion zur Detektion von möglichen Ausnahmefällen aufgerufen. Der Vorteil ist die schnelle Umsetzung bei der Entwicklung eines Simulationsmodells für ein SEWS. Ein Nachteil ergibt sich, wenn dabei relevante Zustandsänderungen übersehen werden. Weiterhin wird hierdurch die Anforderung der Unabhängigkeit und Erweiterbarkeit vernachlässigt (vgl. Abschnitt 3.2.2).

Die zweite Variante verwendet eine berechnete feste Zeitspanne Δt_{Det} zwischen Detektionszeitpunkten. Dabei ergibt sich ein Problem, welches anhand der Abbildung 4.9 geschildert werden kann. Es handelt sich um die Darstellung der Bestandsveränderungen eines beliebigen Puffers über der Simulationszeit. Bei einer angenommenen festen Zeitspanne von fünf Minuten werden mehrere *Peaks* als Ausnahmefall detektiert, ein anderer bleibt unberücksichtigt. Dies macht deutlich, dass die Dauer $\bar{t}_{Exc,i}$ eines Ausnahmefalls i bei der Detektion ebenfalls eine Rolle spielt. Dabei muss bei der Festlegung einer festen Zeitspanne in Kauf genommen werden, dass Ausnahmefälle unter Umständen nicht detektiert werden, wenn für diese $\bar{t}_{Exc} \leq \Delta t_{Det}$ gilt.

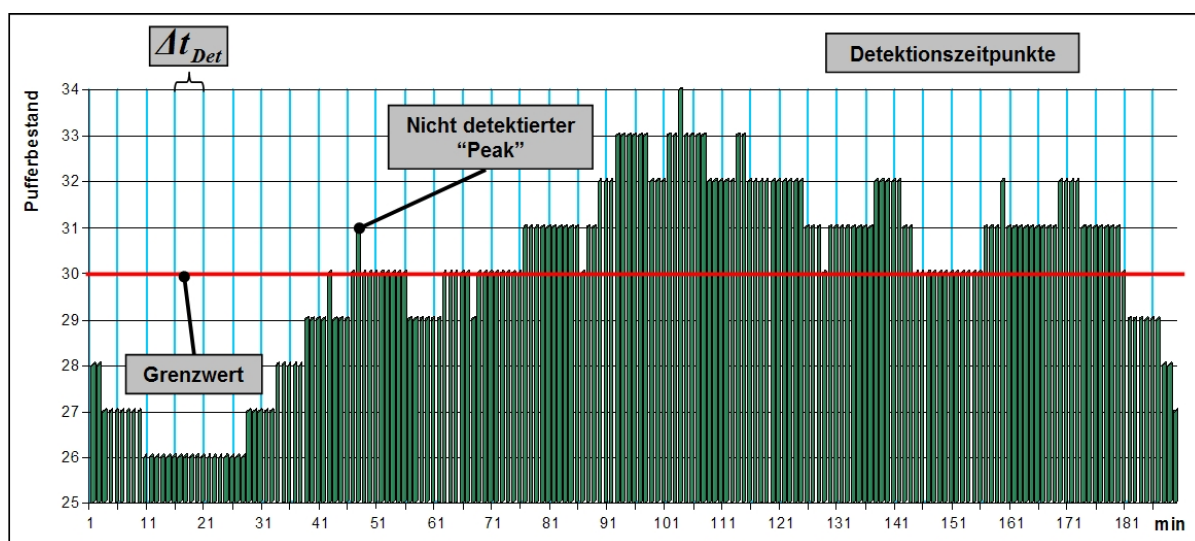


Abb. 4.9: Bestandsverlauf eines Puffers mit Grenzwertüberschreitung

Die Festlegung einer festen Zeitspanne kann *a priori* erfolgen, wenn diese bekannt oder in einer bestimmten Größe gewünscht ist. Ist dies nicht der Fall, besteht die Möglichkeit einer Festlegung über die Analyse der *Zwischenankunftszeiten der Zustandsänderungen* Z^* . Im Bereich der Materialflusslehre ist unter der *Zwischenankunftszeit (ZAZ)* im Allgemeinen der Zeitraum zwischen den Ankünften zweier aufeinanderfolgender Aufträge oder Fördereinheiten zu verstehen.

Die Bestimmung dieser ZAZ kann über die Analyse der Betriebsdaten erfolgen, wozu die relevanten Zustandsänderungen Z^* im System direkt oder indirekt messbar sein müssen. Ist dies

nicht der Fall, so sind hierfür repräsentative Simulationsläufe durchzuführen, die Rückschlüsse auf die ZAZ der Zustandsänderungen Z^* ermöglichen. Dabei wird für jede Zustandsänderungsart i die ZAZ t_i gemessen und statistisch erfasst. Die Ergebnisse dieser Analysen sind in Abhängigkeit des Simulationsmodells höchst unterschiedlich.

Abbildung 4.10 zeigt eine Prinzipskizze der statistischen Verteilung bzw. des Histogramms der ZAZ der Z^* eines beliebigen Materialflusssystemes. Die einzelnen Intervalle stellen die Klassierung der Zwischenankunftszeiten Zustandsänderungen Z^* dar, während das Histogramm die relative bzw. die Verteilungsfunktion die kumulative Häufigkeit der ZAZ aufzeigt.

Um sicher zu stellen, dass auch wirklich alle Zustandsänderungen berücksichtigt werden, die zu Ausnahmefällen führen können, müsste anhand dieser Analyse die minimale ZAZ über alle erfassten ZAZ der Zustandsänderungen Z^* hinweg verwendet werden. Es ist leicht einzusehen, dass die Performance der Simulationsläufe mit abnehmender minimaler ZAZ leidet. Weiterhin ist die tatsächliche Verwendung dieser minimalen Größe in der Praxis fraglich, da folgende Zusammenhänge im Rahmen dieser Arbeit nicht nachgewiesen werden konnten:

- Die Rückführung der aufgetretenen Ausnahmefälle auf die Verwendung der minimalen ZAZ. Es könnte ja auch die Verwendung einer der nächstgrößeren ZAZs zur vollständigen Ausnahmefalldetektion ausreichend sein.
- Die praktische Relevanz eines Ausnahmefalles, dessen Grenzwert nur für die Zeitspanne der minimalen ZAZ auftritt. In den meisten Fällen realer Materialflusssysteme sind beispielweise Überschreitungen von Grenzwerten bei Ausnahmefällen im Sekundenbereich nicht relevant bzw. können vernachlässigt werden. Die zeitliche Dauer eines Ausnahmefalles spielt in der Praxis also ebenfalls eine Rolle.

Diese Problemstellungen bzw. Zusammenhänge sind interessante Fragestellungen für künftige Untersuchungen, um die Detektion von Ausnahmefällen mittels der Verwendung von ZAZ zu verbessern. Im Rahmen dieser Arbeit stellte sich nach mehrfachen Auswertungen die Verwendung eines Vertrauens- bzw. Konfidenzbereiches von $P = 75\%$ bis $P = 90\%$ als sinnvoll heraus. Der Begriff ist der schließenden Statistik entliehen und bezeichnet den Bereich, in dem sich der wahre Parameterwert mit der Wahrscheinlichkeit $P = 1 - \alpha$ befindet, wobei α die Irrtumswahrscheinlichkeit darstellt ([AF05]). In unserem Fall handelt es sich hierbei um den Bereich, in dem sich eine geeignete minimale ZAZ befindet. Dabei werden überwiegend alle relevanten Ausnahmefälle detektiert. Die restlichen Ausnahmefälle konnten vernachlässigt werden, da sie auch in der Praxis keine Rolle spielten.

Zum Abschluss dieses Abschnitts wird noch die Vorgehensweise an den so bestimmten Detektionszeitpunkten erläutert. Dabei ermöglicht der rekursive Aufbau und die Baumstruktur der Berechnungsregeln für Kennwerte eine einfache Implementierung mittels Quellcode im Simulationsmodell. Die notwendigen Attribute der Simulationsentitäten werden durch die Blätter dieser Baumstruktur repräsentiert. Deren Werte werden ermittelt und zur Kalkulation herangezogen. Anschließend werden die berechneten Kennwerte auf die Verknüpfung mit einem Ausnahmefall und einer eventuellen Überschreitung dessen definierter Grenzwerte hin untersucht.

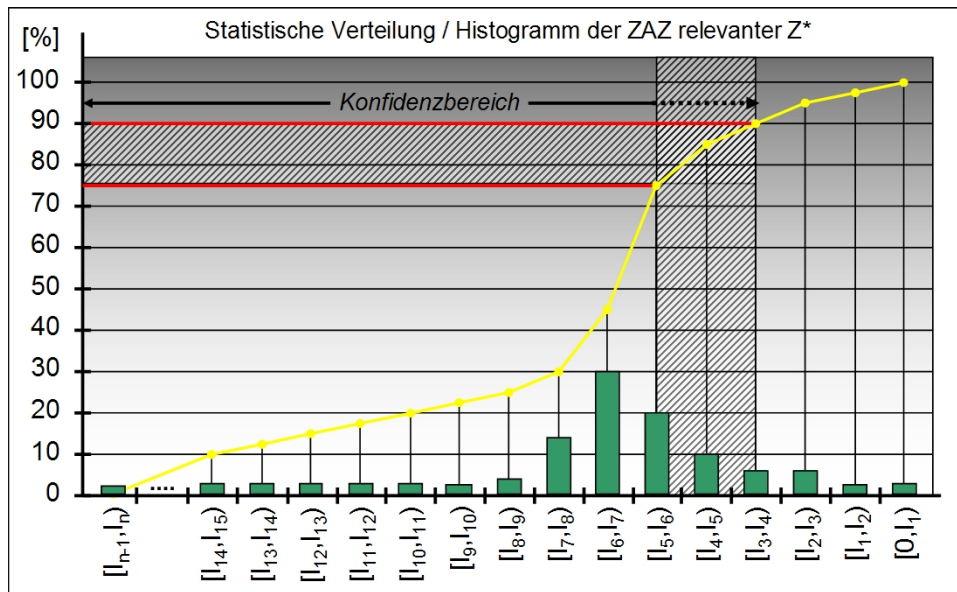


Abb. 4.10: Prinzipskizze der statistischen Verteilung der ZAZ relevanter Z*

Die Darstellung in einem entsprechenden Ausgabeformat zur Weiterverarbeitung im SEWS-Framework spielt ebenfalls eine große Rolle. Da die Daten der Berechnungsregeln und Ausnahmefälle in der SEWS-internen XML-Datenbank verwaltet werden (vgl. Abschnitt 4.1.4), genügt die Angabe der Paarung *ID* (Ausnahmefall bzw. Berechnungsregel) und *Kennwert*. Dies kann wiederum im XML-Format geschehen, welches vom Framework leicht verarbeitbar ist.

Die in dieser Arbeit dargestellte Vorgehensweise zur Beschreibung und Detektion von Ausnahmefällen ist auch für weitergehende Fragestellungen interessant. Zum einen ermöglicht sie eine flexible und simulatorunabhängige Reportgenerierung. Zum anderen können bei Simulationsexperimenten interessierende Vorgänge wie bspw. Stauungen oder Kapazitätsschwankungen untersucht werden. Im Vergleich dazu ist die manuelle Bestimmung schwierig und zeitaufwändig, da es bei Simulationsläufen über längere Simulationszeiträume hinweg beinahe unmöglich ist, alle entsprechenden Zeitpunkte herauszufiltern. An dieser Stelle kann die vorgestellte Vorgehensweise einen Beitrag leisten.

Eine weitere Problematik bei der Prognose mittels Simulation besteht in der Notwendigkeit einer Modellaktualisierung. Hierauf wird im folgenden Abschnitt eingegangen.

4.3.3 Problem der Prognosegenauigkeit

Bei der Verwendung der Simulation zur Prognose ergibt sich die Problematik der Beurteilung der Prognosegenauigkeit. Davon ist die Akzeptanz eines SEWS beim Einsatz in der Praxis abhängig (vgl. Abschnitt 3.2.1). Das Haupteinsatzgebiet der Simulation in der Automobilindustrie liegt sowohl in der Planung von Produktionsanlagen, Logistikbereichen und -konzepten

als auch in der operativen Produktionsplanung und -steuerung (vgl. Abschnitt 2.2). Zur Problematik der Prognosegenauigkeit findet sich im Bereich von Produktion und Logistik keine einschlägige Literatur. Die Thematik wird lediglich im Bereich der Wettervorhersage ausgiebig diskutiert ([Wal01],[HV01]).

Die Einflussfaktoren der Prognosegenauigkeit im Bereich von Produktion und Logistik können auf einige Schwerpunkte zurückgeführt werden:

1. **Datenqualität**

Einen großen Einfluss auf die Genauigkeit hat die Qualität der Daten. Dies hat verschiedene Ursachen. Trotz der heutigen technischen Möglichkeiten herrschen in vielen Industrieunternehmen noch historisch gewachsene Informationssysteme vor (vgl. Abschnitt 4.1.1). Weiterhin werden verschiedene Daten nicht oder nicht regelmäßig gepflegt, wobei die menschliche Komponente eine große Rolle spielt. Aus diesem Grund müssen bei gewöhnlichen Simulationsstudien vor der Modellerstellung Interviews mit den betreffenden Personen durchgeführt werden. Die zu verwendenden Informationssysteme beinhalten teilweise nicht die erforderlichen Daten oder die Frequenz der Datenaktualisierung in diesen Systemen ist zu gering ([FR04]).

2. **Modellgenauigkeit**

Bei der Modellierung können Fehler auftreten. Dabei können hinterlegte Statistiken nicht korrekt sein, es werden bestimmte Zusammenhänge nicht berücksichtigt oder sind schlicht unbekannt. Dieser Problematik kann nur durch eine sehr genaue und umfassende Analyse des Materialflusssystems und der Einhaltung der Sorgfaltspflicht entgegen getreten werden. Hierbei spielen wiederum Erfahrungen im Bereich der Simulation eine Rolle, da in diesem Zusammenhang hohe Anforderungen an den Simulationsexperten gestellt werden (vgl. Abschnitt 2.1.3).

3. **Modellaktualität**

Auch die sorgfältigste Analyse des Materialflusssystems garantiert nicht, dass Unwägbarkeiten während des Betriebes eines SEWS auftreten können. In diesem Fall ist das Simulationsmodell eines SEWS permanent anzupassen und zu verbessern. Zu solchen Unwägbarkeiten gehören Veränderungen im Materialfluss, die den Informationssystemen nicht zurückgemeldet, oder die nicht eingepflegt werden. Weicht die Prognosegenauigkeit verstärkt von den üblichen Ergebnissen ab, ist dies ein Hinweis auf eine Veränderung im realen System, die sich nicht in den simulationsrelevanten Daten widerspiegelt bzw. in den betreffenden informationssystemen nicht aktualisiert worden sind.

Die Reichweite der Plandaten (vgl. Abschnitt 4.1.1), die aus den Informationssystemen gewonnen werden können, beschränkt die Länge des möglichen Prognosezeitraumes. Ein Beispiel sind etwa Produktionsplanungszeiträume von einer Woche, einem Tag oder gar nur einer Schicht. Vor allem in der Teilefertigung wird die Maschinenbelegung erst zu Beginn einer Schicht durchgeführt. Das bedeutet, dass ein SEWS eine Prognose ohne negative Beeinflussung der Prognosegenauigkeit nur für die Dauer einer Schicht vornehmen kann.

An dieser Stelle stellt sich die Frage, wie die Prognosegenauigkeit der Simulation im Rahmen von SEWS bewertet werden kann. Zum einen ist ein Vergleich der Simulationsergebnisse mit

den Daten der Realität durchführbar. Dies bezieht sich auf bestimmte Produktionsziele, die auch in der Simulation erreicht wurden, wie Auslastungsgrade von Werkern und Maschinen oder Pufferbestände. Es besteht oftmals die Problematik, dass die mit Hilfe der Simulation ermittelten Kennwerte nicht oder nur mit Schwierigkeiten mit den Realdaten abgeglichen werden können.

Da Messpunkte in SEWS eine wichtige Rolle spielen und ohnehin zur Initialisierung aus den existierenden Informationssystemen gewonnen werden müssen, stellt die Untersuchung der Messpunktabweichung von Produkten eine genauere und auch für kurzfristige Zeiträume geeignetere Methode zur Beurteilung der Prognosegenauigkeit als der Vergleich der Simulationsergebnisse mit den Daten der Realität dar. Hierzu werden mit Hilfe des Simulationsmodells auf Basis von Historiedaten Prognosen erzeugt, die mit den entsprechenden Vergangenheitsdaten verglichen werden. Für jeden Messpunkt i , mit $i = 1, \dots, n$, und jedes Produkt j , mit $j = 1, \dots, m$, des Vergleichszeitraumes existiert der reale Zeitstempel $t_{i,j}^r$ und der prognostizierte Zeitstempel $t_{i,j}^p$.

Zur Bewertung der Prognosegenauigkeit bzw. des Prognosefehlers können beispielsweise die folgenden üblichen Verfahren bzw. Masszahlen herangezogen werden ([Alb05]):

- Mittlere absolute Abweichung (engl.: Median Absolute Deviation, MAD)
- Fehlersumme (engl.: Error Total, ET)
- Mittlerer prozentualer Fehler (engl.: Mean Percentage Error, MPE)
- Mittlerer absoluter prozentualer Fehler (engl.: Mean Absolute Percentage Error, MAPE)
- Mittlerer quadratischer Fehler (engl.: Mean Squared Error, MSE)
- Quadratwurzel des mittl. quadratischen Fehlers (engl.: Root Mean Square Error, RMSE)

Je nach der Ausprägung der einzelnen Werte können Rückschlüsse auf die notwendigen Veränderungen im Modell oder den Eingangsdaten vorgenommen werden. Die Gestaltung einer allgemeingültigen und automatisierten Anpassung in Abhängigkeit der berechneten Werte ist fraglich. Um die Prognosegenauigkeit zu erhöhen, muss das bestehende Simulationsmodell, die Eingangs- und Plandaten untersucht und die entsprechenden Parameter angepasst werden. Eine vollständige Automatisierung wäre anzustreben. Dies ist allerdings nicht Ziel dieser Arbeit und soll späteren Forschungsarbeiten überlassen werden. Bei einer mangelnden Prognosegenauigkeit wird auf die manuelle Überprüfung und Anpassung zurückgegriffen.

Die Erreichung einer 100 %igen Prognosegüte ist reine Utopie, da es sich bei der Simulation immer um eine Abstraktion der Realität handelt und selbst bei der größten Sorgfalt nicht alle Aspekte der Realität berücksichtigt werden können. Weiterhin ist die Angabe einer erreichbaren prozentualen Prognosegüte nicht sinnvoll, da diese beliebig schlecht sein kann. Aus diesem Grund ist es auch nicht sinnvoll, Maschinenausfälle zu prognostizieren. Die Angabe einer prozentualen Wahrscheinlichkeit eines Maschinenausfalles zu einem bestimmten Zeitpunkt in der Zukunft wäre in der Praxis nicht vermittelbar. Es können aber Aussagen gemacht werden, in welchen üblichen Bereichen sich die absolute und standardisierte Messpunktabweichung des

Simulationsmodells befindet und damit eine qualitative Abschätzung über die Prognosegüte getroffen werden kann.

4.4 Zusammenfassung des Kapitels

In diesem Kapitel wird auf die Prognose künftiger Systemzustände im Rahmen Simulationsbasierter Frühwarnsysteme eingegangen. Hierzu beschäftigt sich ein erster Abschnitt mit der Datenbeschaffung und -verwaltung, wobei zunächst eine Unterscheidung zwischen Simulations- und SEWS-relevanten Daten erfolgt. Anschließend werden Kriterien zur Systemauswahl vorgestellt, die im Zusammenhang der Datenbeschaffung bei der Auswahl geeigneter Informationssysteme und Datenquellen Anwendung finden. Weiterhin wird die SEWS-interne Datenverwaltung mit XML und der Einsatz einer XML-Datenbank erläutert, wonach die eigentliche Datengewinnung aus originären Datenquellen vorgestellt wird.

Ein weiterer Abschnitt beschreibt die Verwendung der Simulation als eingebettete Applikation, wobei auf die Anforderungen an SEWS eingegangen wird, die durch die Simulation beeinflusst werden. Es erfolgt die Darstellung der relevanten Eigenschaften von Simulatoren und die an Simulationsmodelle zu stellenden Anforderungen.

Im letzten Abschnitt dieses Kapitels wird die Simulation als Prognosewerkzeug betrachtet. Dabei geht es um die Problematik der Initialisierung von Simulationsmodellen und der technischen Umsetzung der Detektion von Ausnahmefällen. Zum Abschluss werden sowohl die Möglichkeiten zur Modellaktualisierung als auch das Problem der Prognosegenauigkeit diskutiert.

Dieses Kapitel geht auf die technische Umsetzung von SEWS und die Realisierung der Prognose möglicher Ausnahmefälle im Bereich von Produktion und Logistik ein. Dabei werden Fragestellungen des vorangehenden Kapitels zur Anbindung heterogener Datenquellen und der internen Datenverwaltung Simulationsbasierter Frühwarnsysteme aufgegriffen. Weiterhin wird die Einbindung der Simulation, der hierfür notwendigen Module und die Initialisierung von Simulationsmodellen dargestellt. Auch die Prognose und Detektion von Ausnahmefällen sind diskutierte Themenschwerpunkte. Die erläuterte Vorgehensweise ermöglicht eine flexible und simulatorunabhängige Reportgenerierung. Darüber hinaus ist sie zur automatisierten Untersuchung und Filterung interessanter Vorgänge wie bspw. Materialflusstauungen oder Kapazitätsschwankungen während der Durchführung von Simulationsexperimenten geeignet.

Neben vielen Antworten werden in diesem Kapitel auch neue und weiterführende Fragen aufgeworfen. Hierzu gehören die im Zusammenhang mit Zwischenankunftszeiten von relevanten Zustandsänderungen Z^* verbundenen Fragestellungen (vgl. Abschnitt 4.3.2.2). Dabei geht es um die Rückführung aufgetretener Ausnahmefälle auf die Verwendung minimaler Zwischenankunftszeiten und die praktische Relevanz von Ausnahmefällen mit sehr kurzen Grenzwertüberschreitungen. Weiterhin ist sowohl die Thematik der Entwicklung und Analyse geeigneter Messverfahren für die Prognosegenauigkeit Simulationsbasierter Frühwarnsysteme als auch der vollständigen Automatisierung einer Modellaktualisierung zur Verbesserung der Prognosegüte interessant für künftige Forschungsarbeiten.

Kapitel 5

Handlungsalternativen im Ausnahmefall

Die Lösung ist immer einfach, man muss sie nur finden.

— Alexander Solschenizyn

Eine wesentliche Funktionalität von SEWS ist die Generierung und Bewertung von Handlungsalternativen als Antwort auf das Erkennen von Ausnahmefällen. Dabei ist es für die Generierung nicht entscheidend, ob die Ausnahmezustände im realen oder simulierten Betrieb detektiert werden.

Hierzu existieren in der Literatur bereits diskutierte Ansätze. Feldmann bspw. verwendet in einem so genannten *Monitoring-Framework* einen vordefinierten Alternativenkatalog ([Fel00]). Solche Monitoringsysteme dienen zur permanenten Erfassung und Überwachung des Zustandes des realen Materialflusssystems (vgl. Abschnitt 3.2.3.2). Auch hierbei werden Handlungsalternativen im Ausnahmefall benötigt. Ein strategisches Frühwarnsystem wird von Barfus diskutiert, bei dem Frühwarnindikatoren verwendet werden ([BTS06]). Hierbei stellt ein Ausnahmefall die Abweichung von einer mit diesen Frühwarnindikatoren verknüpften Entscheidungsregel dar, die wiederum mit bestimmten Gegenmaßnahmen verbunden ist.

All diesen Ansätzen ist gemein, dass für spezifische Problemstellungen angepasste Klassifikationen von Ausnahmefällen Anwendung finden, mit denen über eine Art Handbuch bzw. Alternativensammlung vordefinierte Handlungsalternativen fest verknüpft und mit Hilfe der Simulation bewertet werden. Im Unterschied zu SEWS besteht eine vordefinierte Fehlerdatenbank, deren Anwendung eine Interaktion mit dem Menschen erfordert. Dabei existiert für einen spezifischen Ausnahmefall i eine fest definierte Kriterienmenge $K_i := \{K_i^1; \dots; K_i^m\}$, die mit einer fest definierten Handlungsalternativenmenge $HA_i := \{A_i^1; \dots; A_i^n\}$ verbunden ist. Formal handelt es sich also um eine Funktion f , mit

$$f : K_i \rightarrow HA_i.$$

Die finale Auswahl einer Handlungsalternative aus der Alternativenmenge HA_i bleibt dem Anwender des Systems überlassen.

Die Vorteile dieser Ansätze sind deren Übersichtlichkeit, die einfache Implementierung und Realisierung. Die Nachteile liegen in einer mangelnden Übertragbarkeit auf neue Problemstellungen und Erweiterbarkeit um neue Aspekte oder Komponenten, sowie einer sehr starken Abstraktion der Realität.

Zu den Anforderungen an SEWS gehören die Skalierbarkeit, Adaptionfähigkeit, Standardisierung, Erweiterbarkeit, Wiederverwendbarkeit und Unabhängigkeit. Dies beinhaltet auch die flexible Anpassung an noch nicht berücksichtigter Ausnahmefälle, für die wiederum noch keine vordefinierten Handlungsalternativen existieren.

Bei SEWS geht es um die Beeinflussung prognostizierter oder real aufgetretener Ausnahmefälle. Ziel ist die Minimierung der Grenzwertüberschreitung in Abhängigkeit eines Ausnahmefalls, wodurch völlig unterschiedliche und wechselnde Zielfunktionen zu verarbeiten sind. Im Vergleich hierzu versuchen Simulationsbasierte Optimierungsverfahren konkrete Kennwerte eines Materialflusssystemes zu optimieren und somit das Gesamtsystem zu verbessern. Dies entspricht nicht der Zielsetzung Simulationsbasierter Frühwarnsysteme.

In diesem Kapitel wird auf die Thematik der Handlungsalternativen im Ausnahmefall eingegangen. Hierzu erfolgt in einem ersten Abschnitt die Erläuterung der generellen Vorgehensweise zur Generierung von Handlungsalternativen, wobei der Ablauf der Handlungsalternativengenerierung und die generellen Steuerparameter eines realen Materialflusssystemes vorgestellt werden. Weiterhin wird die Notwendigkeit eines Lösungsverfahrens begründet.

Ein zweiter Abschnitt befasst sich mit der Einordnung und Abgrenzung unterschiedlicher Methoden zur Handlungsalternativengenerierung im Ausnahmefall. Zu Vergleichszwecken werden sowohl die Themen Optimierung und Simulation als auch Technologien wie die Fuzzy-Logik, Künstliche Neuronale Netze, Regelbasierte Systeme und Expertensysteme erläutert.

Zum Abschluss erfolgt die Erläuterung eines im Rahmen dieser Arbeit entwickelten Verfahrens zur Ermittlung und Bewertung von Handlungsalternativen. Hierzu wird die mehrstufige Vorgehensweise begründet und die Darstellung und Repräsentation von Handlungsalternativen in der SEWS-internen Datenverwaltung vorgestellt. Anschließend befasst sich der Abschnitt sowohl mit der zeitlichen und qualitativen Belegungsplanung als auch mit quantitativen Steuerparametern eines Materialflusssystemes und Sonderfällen. Weiterhin erfolgt die Darstellung eines allgemeinen Algorithmus zur Handlungsalternativengenerierung, wonach auf die Thematik der Adaption von Korrelationskoeffizienten an ein Materialflusssystem zur Reduzierung der Anzahl durchzuführender Simulationsläufe eingegangen wird. Zum Abschluss befasst sich der Abschnitt mit der Einführung von Abbruchkriterien, die eine Beendigung des vorgestellten Algorithmus garantieren.

5.1 Generelle Vorgehensweise zur Generierung von Handlungsalternativen

Die Generierung von Handlungsalternativen spielt eine große Rolle bei *Decision-Support-Systemen (DSS)*, deren ursprüngliche Aufgabe zur operativen und strategischen Entscheidungsunterstützung des Managements bzw. Entscheidungsträgern dienen ([TAL05]). Eine ähnliche Funktion wie bei der Anwendung der Simulation in SEWS zur Bewertung generierter Handlungsalternativen haben Modelle bei so genannten *Modellgesteuerten Decision-Support-Systemen (MDSS)* ([Gac04]). Hierzu können sowohl statistische oder finanztechnische Modelle als auch Optimierungs- oder Simulationsmodelle herangezogen werden.

Je nach Einsatzgebiet sind Alternativen im Bereich von Produktion und Logistik während Planungs- und Projektierungsphasen und zur Entscheidungsunterstützung von großer Bedeutung (vgl. Abschnitt 2.1.3). In der Automobilindustrie unterstützt die Simulation mit ihren unterschiedlichen Funktionalitäten kostengünstige „*What-if-Analysen*“ zur Steigerung der Planungsqualität ([Bay03]). Unterschiedliche Planungsszenarien werden entwickelt und mit Hilfe der Simulation bewertet.

Bei allgemeinen Frühwarnsystemen ist die Generierung von Handlungsalternativen nicht das primäre Ziel. Es geht vielmehr um die Warnung vor aufkommenden Gefahren und die Information potentiell Gefährdeter (vgl. Abschnitt 3.1). Dabei handelt es sich um vorher identifizierte Ausnahmen und Gefahren, deren Auftreten überwacht bzw. prognostiziert wird.

Im Gegensatz dazu ist bei Simulationsbasierten Frühwarnsystemen im Bereich von Produktion und Logistik die Definition von Ausnahmefällen flexibel gestalt- und veränderbar. Dadurch wird das feste Zuordnungsschema einer festen Alternativenmenge zu einem Ausnahmefall verlassen. Aufgrund der Komplexität von Materialflusssystemen muss eine Handlungsalternative ausgewählt bzw. generiert werden, die den gegenwärtigen Zustand des realen Systems, sowie dessen Eigenschaften und Plandaten wie beispielsweise Kundenaufträge berücksichtigt.

Dies ist eine komplexe Aufgabe, die in diesem Kapitel dargestellt und für die ein Lösungsvorschlag erarbeitet wird. Der folgende Abschnitt geht zunächst auf den allgemeinen Ablauf der Handlungsalternativengenerierung in SEWS ein. Anschließend werden generelle Steuerparameter realer Materialflusssysteme vorgestellt. Weiterhin wird die Notwendigkeit eines Lösungsverfahrens diskutiert.

5.1.1 Ablauf der Handlungsalternativengenerierung

Abschnitt 3.3.2 beschreibt den Aufbau des Frameworks von SEWS. Hierbei wird die Komponente *Generator of Alternatives (GoA)* eingeführt, deren Aufgabe die Generierung von Handlungsalternativen beim Auftreten von Ausnahmefällen ist. Die zeitliche Einordnung dieses Vorgangs in den Workflow von SEWS wird in Abbildung 3.5 dargestellt.

Die Komponente GoA wird im Ausnahmefall angestoßen, wenn die Analyse der realen Daten oder der Ergebnisse einer Prognose durch den Exception Interpreter mindestens eine Grenzwertüberschreitung eines definierten Typs eines Ausnahmefalls ergibt (vgl. Abschnitt 4.3.2.1). Dabei beinhaltet diese Komponente wiederum mehrere Funktionen, die eine Handlungsalternativengenerierung ermöglichen. Hierzu gehört ein *Manipulator*, der die Steuerparameter der Simulation verändert. Dabei können nur die Umfänge der Steuerparameter verändert werden, die auch im realen System zur kurz- und mittelfristigen Steuerung herangezogen werden können. Hierzu gehören bspw. die Belegungsplanung, die Anzahl eingesetzter Werker oder Schichtdauern.

Der Manipulator generiert in Abhängigkeit des Ausnahmefalls verarbeitbare Werte für die Steuerparameter und greift hierzu auf in der Wissensbasis (Knowledge Base) hinterlegte Regeln und erlerntes Wissen zurück (vgl. Abschnitt 3.3.2). Anschließend werden mit Hilfe der

Komponente Simulation Control neue Simulationsläufe gestartet, deren Ergebnis die neu berechneten Kennwerte und detektierten Ausnahmefälle sind.

Diese Informationen werden durch die Funktion *Analyzer* untersucht. Dabei wird überprüft, ob der die Handlungsalternativengenerierung auslösende Ausnahmefall positiv beeinflusst werden konnte, oder ob eventuell neue Ausnahmefälle durch die Veränderung der Steuerparameter aufgetreten sind. Weiterhin kann ein *Abbruchkriterium* erfüllt sein.

Wurden geeignete Handlungsalternativen gefunden, so werden diese der SEWS-internen Datenverwaltung und dem User-Interface zur Verfügung gestellt. Abbildung 5.1 zeigt die Bestandteile der Komponente GoA.

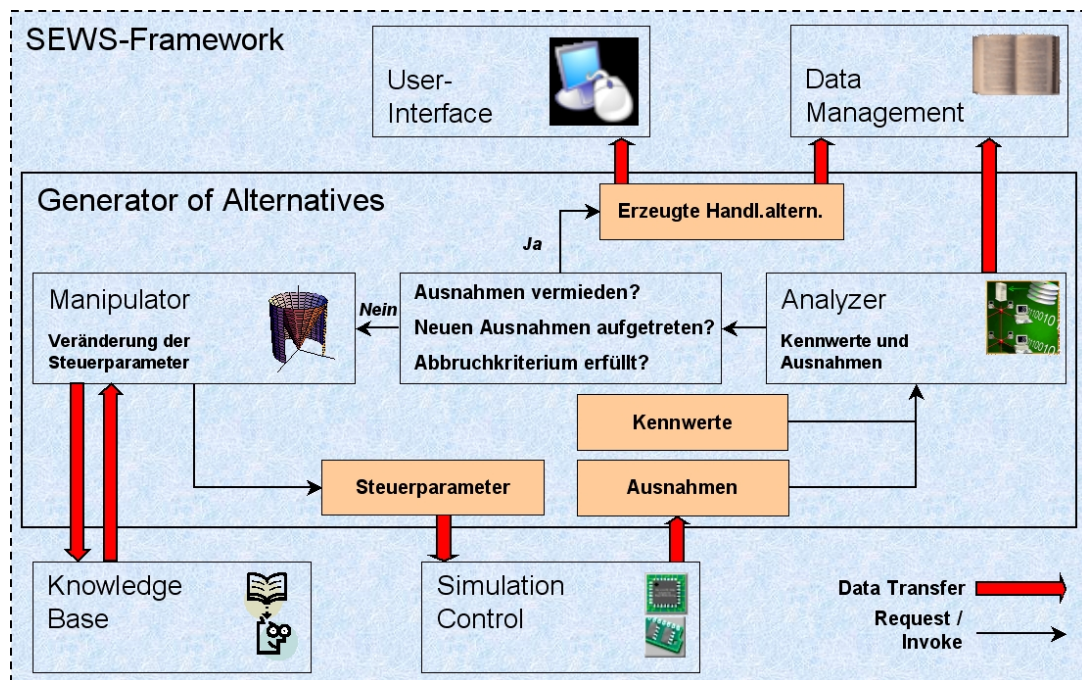


Abb. 5.1: Aufbau der Komponente Generator of Alternatives

5.1.2 Steuerparameter für reale Materialflusssysteme

Die möglichen Handlungsalternativen werden durch bestimmte Parameter vorgegeben, mit denen das System gesteuert werden kann. Grundsätzlich können folgende Parametergruppen für ein Materialflusssystem unterschieden werden (vgl. Abbildung 5.2):

- **Quantitativ:**

Quantitative Steuerparameter sind diskrete oder diskretisierbare Einflussgrößen, die in einem bestimmten Wertebereich variiert werden. Dies ist beispielsweise die Anzahl von bestimmten Werkertypen, Verlängerungen von Schichtzeiten oder quantitativ veränderbare Ressourcen. Die für ein spezifisches Materialflusssystem möglichen quantitativen Steuerparameter sind bei der Etablierung eines SEWS zu berücksichtigen.

- **Qualitativ:**

Qualitative Steuerparameter stellen die Durchführung einer Belegungsplanung und systemspezifische Sonderfälle dar.

Die Belegungsplanung ist ein komplexes Themengebiet und sehr stark abhängig von der Problemstellung. Sie gehört zur Klasse der ganzzahligen und kombinatorischen Optimierungsprobleme und befasst sich mit der optimalen Einplanung von Aufträgen zur Durchführung von Prozessen mit Hilfe bestimmter Ressourcen unter Beachtung vorgegebener Zielsetzungen und Restriktionen ([NM02]). Wird ein Materialflusssystem mit einem SEWS ausgestattet, so muss ein separates Modul existieren, welches im Ausnahmefall neue Belegungspläne mit den sich dadurch ergebenden Restriktionen erzeugt.

Bei systemspezifischen Sonderfällen handelt es sich um Alternativen, die nicht allgemein formulier- und darstellbar sind. Dies können bspw. Ausweichstrategien wie Ersatzmaschinen oder spezielle Werkzeuge sein, die im Vergleich zum üblichen Materialfluss manuell betrieben oder eingerichtet werden müssen und nur in Abhängigkeit spezifischer Ausnahmefälle wirksam werden.

- **Zeitlich:**

Zeitliche Steuerparameter verändern die zeitliche Einsteuerung von Produkten bzw. Aufträgen in das Materialflusssystem. Die Auftragsreihenfolge kann verändert werden, ohne die geplante Belegung von Ressourcen durch diese Aufträge zu beeinflussen. Dadurch ist wiederum eine neue Belegungsplanung mit speziellen Restriktionen erforderlich.

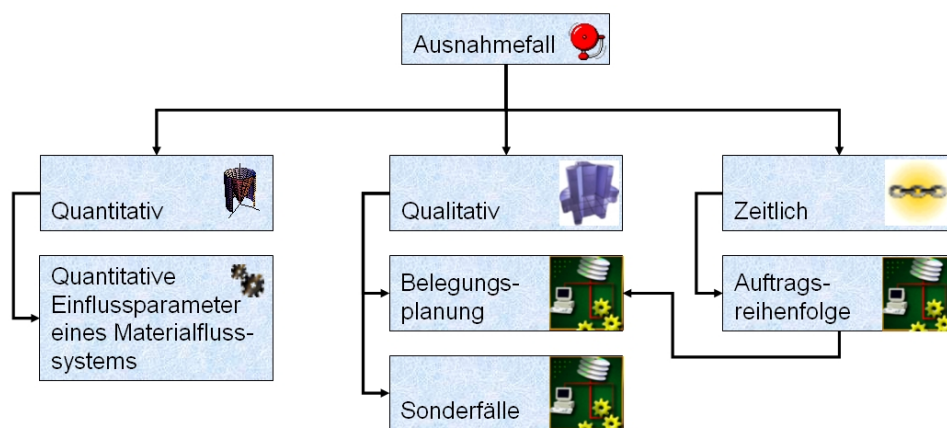


Abb. 5.2: Klassierung möglicher Steuerparameter eines Materialflusssystems

Die Anzahl der möglichen Handlungsalternativen X_{ha} ergibt sich aus

- der Anzahl n^* der diskreten Werte n_i (mit $i = 1, \dots, x$) der Wertebereiche aller x Steuerparameter, mit $n^* = \sum_{i=1}^x n_i$,
- der Anzahl der möglichen Sonderfälle s ,
- der Menge y der möglichen Belegungspläne im Materialflusssystem und
- der Anzahl möglicher Auftragsreihenfolgen z zu

$$X_{ha} = n^* \cdot s \cdot y \cdot z.$$

Bei exemplarischen Untersuchungen in realen Systemen wurden tausende möglicher Alternativen berechnet. Ordnet man jeder Alternative ein Simulationsexperiment zu und führt zur Berechnung der Werte einer Handlungsalternative gegebenenfalls mehrere Simulationsläufe durch, so ist leicht zu erkennen, dass aus Gründen der Reaktionszeit eines SEWS nicht alle Alternativen bewertet werden können. Dies ist auch durch eine Parallelisierung nicht wesentlich zu verbessern. Somit muss eine Reduzierung der Anzahl durchzuführender Simulationsexperimente erfolgen.

Die in Abbildung 5.1 dargestellte zyklische Generierung und Bewertung von Handlungsalternativen ist nicht unbegrenzt möglich, so dass ein oder mehrere geeignete Abbruchkriterien erforderlich sind (vgl. Abschnitt 5.1.1).

Aus all diesen Gründen ist ein Lösungsverfahren notwendig, welches die Steuerparameter eines realen Materialflusssystemes in geeigneter Weise unter Berücksichtigung dieser Umstände verändert und dabei die folgende grundsätzliche Vorgehensweise berücksichtigt:

1. Auswahl einer Untermenge von geeigneten Alternativen aus der insgesamt möglichen Anzahl von Handlungsalternativen X_{ha} .
2. Bildung einer sinnvollen Reihenfolge der Alternativenauswahl aus dieser Untermenge, in Abhängigkeit der Wirksamkeit bezüglich eines Ausnahmefalls.
3. Einführung eines Abbruchkriteriums, um aus Performancegründen die Anzahl durchzuführender Simulationsexperimente weiter zu begrenzen.

Dabei ist nicht das Hauptziel, eine optimale Lösung als Antwort auf einen Ausnahmefall zu finden oder den Ausnahmefall unter allen Umständen zu verhindern. Dies ist in gewissen Situationen gar nicht möglich. In solchen Fällen sollte aber zumindest eine Verbesserung der Situation erreicht werden, so dass sich die Auswirkungen des tatsächlich auftretenden Ausnahmefalls verringern.

Bevor ein solches Lösungsverfahren vorgestellt wird, erfolgt im nachfolgenden Abschnitt eine Einordnung und Abgrenzung der Handlungsalternativengenerierung im Vergleich zu bereits existierenden Ansätzen und Methoden.

5.2 Einordnung und Abgrenzung der Problemstellung

Für die Generierung von Handlungsalternativen wird ein geeignetes Lösungsverfahren benötigt, mit dem zum einen sinnvolle Handlungsalternativen in Abhängigkeit des Ausnahmefalls generiert und zum anderen die Anzahl der Simulationsläufe zur Bewertung der Handlungsalternativen möglichst gering gehalten wird.

In diesem Abschnitt erfolgt zunächst eine Einordnung der Problemstellung zur Generierung von Handlungsalternativen und eine Abgrenzung zu bestehenden Ansätzen. Hierzu gehören im

Folgenden sowohl die Themengebiete Optimierung und Simulation, Fuzzy-Logik und Künstliche Neuronale Netze als auch Regelbasierte Systeme und Expertensysteme.

5.2.1 Optimierung und Simulation

Die generelle Problemstellung der *Simulationsbasierten Optimierung (SO)* liegt in der Kombination von Simulationsmodellen mit einer Optimierungskomponente. Im Vordergrund steht die Bestimmung von Entscheidungsvariablen für die Simulation, die eine Zielfunktion unter der Annahme bestehender Nebenbedingungen minimieren oder maximieren ([NM02]).

Die SO verwendet mit Hilfe eines Optimierungsverfahrens generierte Inputwerte für die Simulation. Mit Hilfe von Simulationsexperimenten wird deren Lösungsgüte bewertet bzw. evaluiert, indem die Parameterwerte für eine zu optimierende Zielfunktion geliefert werden ([PGN06]). Auf Basis dieser Bewertung und den Ergebnissen historischer Bewertungen, generiert die Optimierung neue Inputwerte, die eine Verbesserung der Zielfunktion erreichen sollen. Dieser zyklische Prozess der Inputgenerierung und -bewertung wird iterativ durchgeführt, bis ein bestimmtes Abbruchkriterium erfüllt ist ([NM02]). Dabei kann es sich um eine maximale Anzahl von Zyklen oder eine definierte Ergebnisgüte der Zielfunktion handeln.

In Silver finden sich bezüglich der Verknüpfung von Optimierungsverfahren und der Simulation verschiedene charakteristische Kriterien ([FI02]). Dabei handelt es sich um kausale und temporale, sowie zielbezogene und implementierungsbedingte Kriterien, die zum einen die gegenseitige Abhängigkeit bzw. den zeitlichen Ablauf und zum anderen die Art der Verknüpfung der beiden Komponenten Simulation und Optimierung beschreiben.

Die generelle Architektur eines traditionellen SO-Systems besteht aus einer abstrakten Black-Box, deren interne Prozesse hinter einer Datenschnittstelle verborgen sind ([PGN06]). Dadurch wird ein normaler Anwender unter Verwendung einer geeigneten Benutzerschnittstelle nicht mit der Komplexität eines SO-Systems konfrontiert. Eine weitere Komponente ermöglicht die Konfiguration und die Kontrolle der SO-internen Prozesse, die von einem Spezialisten betreut wird. Abbildung 5.3 stellt diese generelle Architektur noch einmal übersichtlich dar.

Der Typ des zu lösenden Optimierungsproblems bedingt dabei die einsetzbaren Lösungsmethoden eines SO-Systems, wobei sich der Typ aus mathematischer Sicht aus der Kombination verschiedener Zielfunktionen und Restriktionen ergibt. Im Bereich der Simulationsbasierten Optimierung kommen folgende grundsätzliche, so genannte *Instanzbasierte* Methoden in Frage ([FAG05]):

- Ranking and Selection (R&S) ([KN03],[Fu05])
- Response Surface Methodology (RSM) ([Bar05])
- Gradientenbasierte Verfahren ([FAG05])
- Random Search ([And05])
- Sample Path Optimization (SPO) ([FAG05])
- Metaheuristiken ([FAG05])

Neuere und vielversprechende Methoden ergeben sich im Bereich der *Modellbasierten Methoden*, die im Gegensatz zu den *Instanzbasierten* Methoden nicht explizit abhängig von einer aktuellen Lösung sind. Sie verwenden eine Wahrscheinlichkeitsverteilung, um eine Abschätzung über die Positionen der besten Lösungen in einem Lösungsraum abzugeben.

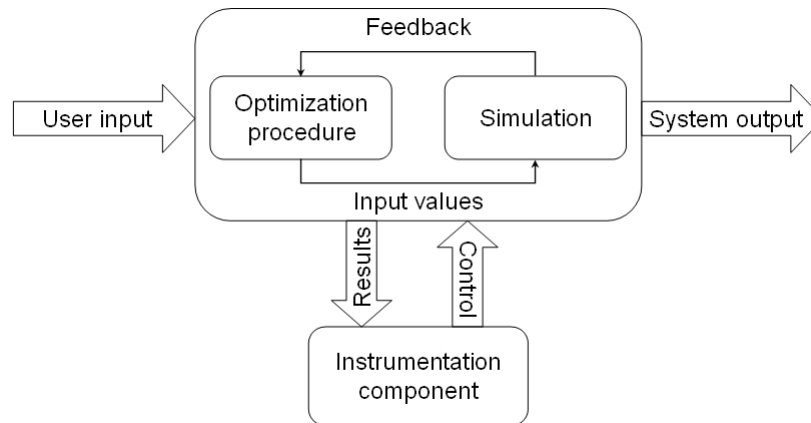


Abb. 5.3: Generelle Architektur eines traditionellen SO-Systems ([PGN06])

5.2.2 Fuzzy-Logik und Künstliche Neuronale Netze

Die Fuzzy-Logik und Künstliche Neuronale Netze können auf unterschiedliche Problemstellungen angewendet werden. Man findet Sie in verschiedenen Einsatzbereichen wie etwa der Produktionsplanung, der Steuerungstechnik, dem Operations Research oder zur Prognose bzw. Analyse von Börsenkursen. Dies legt nahe, dass die Fuzzy-Logik und Künstliche Neuronale Netze zur Handlungsalternativengenerierung verwendbar sind. Aus diesem Grund sollen diese Themengebiete im Folgenden näher betrachtet werden.

Fuzzy-Logik

Die *Fuzzy-Logik* ist eine Theorie zur Verallgemeinerung der zweiwertigen Booleschen Logik, die zur Darstellung unscharfen Wissens entwickelt worden ist. Basierend auf der Fuzzy-Set-Theorie, die als unscharfe Mengenlehre bezeichnet wird, werden nicht nur Werte wie *WAHR* oder *FALSCH* verarbeitet, sondern auch unscharfe Aussagen wie *WENIGER*, *MEHR* oder *VIEL*. Die Haupteinsatzgebiete der Fuzzy-Logik liegt in Bereichen wie der Automatisierungstechnik, Betriebswirtschaft, Medizintechnik, Konsumelektronik oder Regelungstechnik. Dabei wird die so genannte *Fuzzy-Control* sowohl zur Steuerung von Maschinen und Robotern, handelsüblichen Haushaltsgeräten als auch zur Regelung von U-Bahnen, zur Prognose der zukünftigen Last in Routern und Mobilfunk-Stationen, zur Steuerung automatischer Automobilgetriebe, in Alarmsystemen oder im Bereich der Brandmeldetechnik verwendet ([EVW05],[Dri01],[CFF04]).

Um die Fuzzy-Logik einzusetzen, muss keine exakte mathematische Formulierung eines Problems vorliegen. Es genügt eine verbale Beschreibung des Sachverhaltes mit unscharfen Begriffen, die in fest definierte und verarbeitbare Werte für eine Steuerung umzuwandeln sind. Diese Werte werden aus Erfahrungen, Beobachtungen und empirischen Untersuchungen gewonnen. Die Grundlage bilden *Unscharfe Mengen*, bei denen im Gegensatz zur traditionellen Mengenlehre Elemente in einer Menge auch nur zu einem bestimmten Grad enthalten sein können, der mit Hilfe einer *Zugehörigkeitsfunktion* (Anm.: Der so genannten *Fuzzy-Funktion*) beschrieben wird. Mit Hilfe von Fuzzy-Funktionen sind anschließend aus gewichteten Aussagen wie *weniger* oder *mehr* konkrete Zahlenwerte berechenbar, wofür eine große Anzahl möglicher Verfahren existiert.

Künstliche Neuronale Netze

Ein *Künstliches Neuronales Netz (KNN)* stellt im Grunde genommen eine Graphen mit einer bestimmten Menge M an Neuronen dar, die primitiven Verarbeitungseinheiten mit geringem Funktionsumfang entsprechen. Sie arbeiten selbständig und unabhängig voneinander, ohne externe Kontroll- und Steuerungsmechanismen und entsprechen mathematischen Schwellenfunktionen ([Cal03]). Neuronen können also als logische Funktionen (Anm.: Bool'sche Funktionen) charakterisiert werden, die mehrere Eingangssignale empfangen und ein Ausgangssignal senden. Dieses wird bei der Überschreitung eines bestimmten Schwellenwertes durch die Eingangssignale erzeugt ([RN03]).

Die Verbindungen zwischen den Neuronen entsprechen Schnittstellen, über die Informationen ausgetauscht werden können und denen Gewichte zugeordnet sind. Diese Gewichte stellen das in KNN gespeicherte Wissen dar, da sie in Abhängigkeit eines Eingabemusters durch parallele Verarbeitung ein Ausgabemuster erzeugen. Das Lernen kann auf zwei grundsätzliche Arten geschehen. Zum einen können neue Verbindungen zwischen Neuronen geschaffen werden, zum anderen sind Gewichte veränderbar. Es existiert eine große Anzahl möglicher Lernregeln, die unterschiedliche Fähigkeiten bei der Lösung einer Aufgabenstellung aufweisen, worauf an dieser Stelle nicht genauer eingegangen wird.

Mit Hilfe von KNN sind bspw. Belegungsplanungsprobleme lösbar, indem historische Produktionskonstellationen und Belegungspläne analysiert und durch das KNN ein Muster und somit Regeln zur Belegungsplanung abgeleitet werden. Dadurch ist es möglich, für neue Produktkonstellationen neue Belegungspläne zu generieren. Allerdings setzt diese Vorgehensweise ein weitgehend gleiches Auftragspektrum voraus ([Loh94]). Weiterhin sind KNN zur Lösung von Optimierungsproblemen einsetzbar, indem eine Zielfunktion und Nebenbedingungen mit Hilfe eines KNN formuliert werden.

5.2.3 Regelbasierte Systeme und Expertensysteme

Regelbasierte Systeme bestehen aus einer Menge von Daten bzw. einer Datenbank von bestimmten Fakten, einer Menge von Regeln und einem Kontrollsystem mit Regelinterpretern, die auch als *Business-Rule-Engine* bezeichnet wird ([BKI06]). Es handelt sich im Allgemeinen um ein Wissensbasiertes System, in dem ein regelbasiertes Schließen stattfindet.

Die Regeln werden in einen Prämissen- (*WENN*) und einen Konklusionsteil (*DANN*) gegliedert, wobei die Verwaltung in einem so genannten *Business-Rule-Repository* erfolgt. Die Aufgaben des Kontrollsystems sind die Identifikation geeigneter Regeln, das Anwenden von ausgewählten Regeln und die Aktualisierung der Faktendaten. Regelbasierte Systeme sind die Grundlage von Expertensystemen, die zur Lösung oder Bewertung bestimmter Problemstellungen dienen und ein Teilgebiet der künstlichen Intelligenz darstellen.

Die Problematik bei Regelbasierten Systemen und Expertensystemen liegt in der Beschreibung bzw. der Notwendigkeit von Beispielmengen. Wird ein reines Regelbasiertes System verwendet, so ist eine Beschreibung von Regeln durch einen menschlichen Experten notwendig. Mit Hilfe dieser Regeln sollen Schlussfolgerungen für konkrete Situationen gezogen werden. Dies ist allerdings nicht immer ohne menschliches Eingreifen garantiert. Findet hierbei ein Lernprozess Anwendung, so müssen wiederum repräsentative Beispiele in ausreichender Menge gefunden und zur Verfügung gestellt werden.

5.2.4 Einordnung der Handlungsalternativengenerierung

An dieser Stelle erfolgt eine Einordnung und Abgrenzung der Handlungsalternativengenerierung in Simulationsbasierten Frühwarnsystemen im Vergleich mit den zuvor erläuterten potenziellen Lösungsverfahren.

Gemeinsamkeiten mit der SO finden sich in der Verwendung der Simulation in Form der generellen Architektur einer abstrakten Black-Box (vgl. Abb. 5.3), da die internen Prozesse der Simulation hinter einer Datenschnittstelle verborgen sind. Im Falle von SEWS wird die Simulation mit Hilfe eines Web Service gekapselt, welcher die SEWS- und simulationsrelevanten Daten in ein simulatorspezifisches Format transformiert. Weiterhin ist die Handlungsalternativengenerierung als sequentielle Verknüpfung mit einem vollautomatischen Datenaustausch entsprechend der Kriterien von SO-Verfahren charakterisierbar (vgl. Abschnitt 5.2.1). Darüber hinaus werden wie bei SO-Verfahren Inputwerte durch den Generator of Alternatives für die Simulation generiert (vgl. Abschnitt 5.1.1).

Im Vergleich zur Simulationsbasierten Optimierung liegt ein Unterschied in der Definition einer festen Zielfunktion, die in einem SEWS erst nach aufgetretenen oder prognostizierten Ausnahmefällen bestimmt werden kann. Abhängig von der Definition des betreffenden Ausnahmefalls sind unterschiedliche Werte zu minimieren oder zu maximieren. Weiterhin bedingen sich die Ausnahmefälle gegenseitig, da bspw. die Veränderung eines bestimmten Parameters zwar einen Ausnahmefall erfolgreich vermeidet, allerdings einen anderen neu verursacht.

Die SO verwendet hierzu Optimierungsverfahren, um die optimale Lösung für einen Parametersatz zu finden. Dieser Grundgedanke wird für SEWS zwar übernommen, aber es wird aufgrund einer variablen Zielfunktion und aus Performancegründen nicht zwangsläufig die optimale Lösung angestrebt. Deshalb sind reine SO-Verfahren nicht auf die Problemstellung der Handlungsalternativengenerierung anwendbar.

Die Fuzzy-Logik und KNN sind in der Lage, unscharfes Wissen zu verarbeiten und bisher nicht aufgetretene Problemfälle zu lösen. Dies entspricht der Problemstellung bei der Handlungsalternativengenerierung in SEWS, da aufgetretene oder prognostizierte Ausnahmefälle und entsprechende Handlungsalternativen abhängig vom gegenwärtigen Zustand des realen Systems sind. In KNN bspw. könnte ein Ausnahmefall und der gegenwärtige Systemzustand das Eingangsmuster darstellen, wofür das KNN ein entsprechendes Ausgangsmuster (Anm.: Die Vermeidung des Ausnahmefalls) generiert.

Die Differenzen zwischen der Fuzzy-Logik bzw. KNN und der Handlungsalternativengenerierung in SEWS ergeben sich wiederum durch das Ziel, eine optimale Lösung für eine feste Zielfunktion zu finden. Eine weitere Problematik ergibt sich im praktischen Einsatz. Zur Verwendung dieser beiden Technologien müsste zunächst eine ausreichend große Trainingsdatenmenge über das reale System gewonnen werden. Weiterhin müssten Trainingsdaten für ein KNN Eingangs- und Ausgangsmuster beinhalten und für mögliche Ausnahmefälle die Lösungen bekannt sein. Dies stellt einen erheblichen Aufwand zur Datenerhebung dar, der in der Praxis nicht zu leisten ist.

Problematisch ist weiterhin die Berücksichtigung neuer Arten von Ausnahmefällen, die zuvor nicht detektiert wurden bzw. werden sollten. Es ist nicht möglich, ein bestehendes KNN nachträglich zu trainieren. Es muss ein vollständig neuer Lernvorgang mit neuen Trainingsdaten durchgeführt werden. Auch dies ist aufgrund der Anforderungen Erweiterbarkeit, Skalierbarkeit und Adaptionfähigkeit an ein SEWS keine geeignete Lösung.

Regelbasierte Systeme bzw. Expertensysteme verwenden so genannte *WENN- / DANN*-Regeln zur Problemlösung. Wie bei SEWS ergibt sich ein Klassifikationsproblem, um anschließend eine geeignete Lösung zur Ausnahmefallvermeidung zu finden und eine Entscheidungsunterstützung zu ermöglichen. Der Unterschied liegt in dem Zeitpunkt der Erzeugung des *WENN*-Teils, der bei SEWS erst in Abhängigkeit des Ausnahmefalls erzeugt werden kann.

Wie bei der Fuzzy-Logik und KNN benötigen Regelbasierte Systeme bzw. Expertensysteme repräsentative und ausreichend große Beispielmengen, was zu den bereits angesprochenen Problemen führt. Ein weiterer Unterschied ergibt sich in dem häufigen manuellen Einschreiten durch einen Experten, wohingegen die Handlungsalternativengenerierung aufgrund der Anforderungen an SEWS weitestgehend automatisch erfolgen soll.

Diese Zusammenhänge werden in Abbildung 5.4 nochmals übersichtlich dargestellt. Es wird deutlich, dass die vorgestellten Lösungsverfahren und Methoden nicht geeignet sind. Aus diesem Grund soll im Folgenden Abschnitt ein geeignetes Verfahren vorgestellt werden.

5.3 Ein Verfahren zur Ermittlung und Bewertung von Handlungsalternativen

Die Ermittlung und Bewertung von Handlungsalternativen ist eine wesentliche Funktionalität Simulationsbasierter Frühwarnsysteme. Dabei ist ein geeignetes Verfahren zu entwickeln, da

die im vorangegangenen Abschnitt vorgestellten Ansätze wie die Fuzzy-Logik oder Künstliche Neuronale Netze keine verwendbare Standardlösung bieten.

Ein solches Verfahren wird in diesem Abschnitt vorgestellt, welches bei der Ausnahmefalldetektion auf Basis von prognostizierten Daten zukünftiger Zustände des realen Systems oder auf Basis der realen Daten zur Anwendung kommt. Dabei wird in diesem Abschnitt zunächst die mehrstufige Vorgehensweise begründet und anschließend die Darstellung von Handlungsalternativen in Verbindung mit der in den Abschnitten 4.1.1 und 4.1.3 dargelegten SEWS-internen Datenverwaltung erläutert. Weiterhin erfolgt eine Betrachtung der qualitativen und zeitlichen Belegungsplanung sowie der quantitativen Steuerparameter und Sonderfälle.

Es wird ein allgemeiner Algorithmus zur Handlungsalternativengenerierung unter Nutzung von Korrelationskoeffizienten skizziert. Dadurch wird eine Reduzierung der Anzahl möglicher Handlungsalternativen bezüglich der vorhandenen quantitativen Steuerparameter und Sonderfälle erreicht. Dabei sind die Korrelationskoeffizienten an ein spezifisches Materialflusssystem zu adaptieren. Zum Abschluss werden entsprechende Abbruchkriterien eingeführt, die eine Beendigung des allgemeinen Algorithmus garantieren und dabei die Performance des SEWS berücksichtigen.

SO-Verfahren	Fuzzy / KNN
<ul style="list-style-type: none"> – Suche nach einer optimalen Lösung – Verwendung einer festen Zielfunktion 	<ul style="list-style-type: none"> – Optimale Lösung einer festen Zielfunktion – Repräsentative / ausreichend große Beispielmenge – Erweiterung um neue Sachverhalte schwierig
<ul style="list-style-type: none"> + Generierung von Inputwerten für die Simulation + Sequentielle Verknüpfung + Vollautomatischer Datenaustausch + Abstrakte Black-Box als generelle Architektur 	<ul style="list-style-type: none"> + Unscharfes Wissen / Ausnahmefall und entspr. Handlungsalternative sind zustandsbedingt + Prognosefähigkeit des Systems + Belegungsplanung für ein Materialflusssystem
<ul style="list-style-type: none"> + WENN- / DANN-Regeln zur Problemlösung + Klassifikationsproblem im Ausnahmefall + Entscheidungsunterstützung durch so genannte Entscheidungsbäume 	Handlungsalternativengenerierung in SEWS
<ul style="list-style-type: none"> – Manuelles Einschreiten durch einen Experten – Repräsentative / ausreichend große Beispielmenge <p style="text-align: center;">Regelbasiert / Expertensysteme</p>	

Abb. 5.4: Differenzen und Gemeinsamkeiten mit anderen Lösungsverfahren

5.3.1 Mehrstufige Vorgehensweise der Handlungsalternativengenerierung

Reale Materialflusssysteme sind sehr komplex und können höchst unterschiedlich aufgebaut sein. Es existiert eine Vielzahl möglicher Steuerparameter, mit denen ein Materialflusssystem gesteuert bzw. beeinflusst werden kann. Aus diesem Grund existiert kein standardisiertes Lösungsverfahren, welches aus dieser Vielzahl an Einflussmöglichkeiten in akzeptabler Rechenzeit geeignete oder gar die geeignetste Lösung für einen Ausnahmefall auswählen kann.

Hierzu wären eine Vielzahl von Nebenbedingungen und Kombinationen möglicher Steuerparameter zu berücksichtigen, die mit Hilfe der Simulation auf ihre Wirksamkeit hin bezüglich eines Ausnahmefalls überprüft werden müssten. Um eine Reduzierung der Anzahl möglicher Handlungsalternativen bereits frühzeitig zu ermöglichen und somit die Performance eines SEWS zu steigern, erfolgt im Rahmen dieser Arbeit eine entsprechende Einteilung der Steuerparameter in drei grundsätzliche Gruppen, die eine zeitliche, qualitative und quantitative Beeinflussung bzw. Vermeidung von Ausnahmefällen unterstützen (vgl. Abschnitt 5.1.2).

Diese Gruppen können mit unterschiedlichen Lösungsansätzen verbunden werden, die auf die jeweiligen Steuerparameter spezialisiert sind. So sind bspw. zur zeitlichen und qualitativen Beeinflussung Algorithmen zur Belegungsplanung verwendbar, die wiederum spezifisch auf ein Materialflusssystem angepasst werden müssen. Dagegen sind zur quantitativen Beeinflussung diskrete Steuerparameter entsprechend des aufgetretenen Ausnahmefalls zu verändern, um eine Vermeidung bzw. Verbesserung der Situation zu erreichen.

In Abschnitt 5.2.4 wird der Unterschied zu SO-Verfahren in der Definition einer möglichen Zielfunktion deutlich. Während bei der Simulationsbasierten Optimierung die Minimierung oder Maximierung eines konkreten Parameters bzw. Kennwertes wie bspw. die Lagerkosten oder die Auslastung von Maschinen im Vordergrund stehen, geht es bei SEWS um die Beeinflussung prognostizierter oder real aufgetretener Ausnahmefälle. Es wird die Minimierung der Grenzwertüberschreitung im Ausnahmefall angestrebt, wodurch völlig unterschiedliche und wechselnde Zielfunktionen zu verarbeiten sind. Ziel ist nicht die Optimierung des bestehenden Materialflusssystems, wie dies bei SO-Verfahren der Fall ist.

Die Wirksamkeit von Alternativen können sich gegenseitig bedingen. Aus diesem Grund kann eine parallele Anwendung bzw. Bewertung von Handlungsalternativen nicht erfolgen. Deshalb wird eine durch die Verwendung einer spezifischen Handlungsalternative herbeigeführte Verbesserung bei der Bewertung der Wirksamkeit einer weiteren Handlungsalternative berücksichtigt. Es wird also eine mehrstufige Vorgehensweise gewählt.

Die primäre Zielsetzung der Handlungsalternativengenerierung liegt nicht in der Lokalisierung einer optimalen Lösung. Aus diesem Grund kann die Reihenfolge der zu betrachtenden Steuerparameter aufgrund praktischer Erwägungen gewählt werden. Die quantitative Veränderung von Steuerparametern wie bspw. die Reduzierung oder Erhöhung von Ressourcen sind kurzfristig nicht möglich bzw. in der betrieblichen Praxis oftmals mit höheren Kosten verbunden. Dagegen sind in einem kurzfristigen Umfeld die Belegungsplanung bzw. Veränderung von Auftragsreihenfolgen mit einer gleichbleibenden Ressourcenanzahl günstigere und einfachere Möglichkeiten. Aus diesem Grund gliedert sich ein entsprechender logischer Ablauf bei Auftreten eines oder mehrerer Ausnahmefälle in die folgenden drei Schritte:

1. Bestimmung der Zielfunktion.
2. Durchführung einer Belegungsplanung.
3. Überprüfung der Wirksamkeit der quantitativen Steuerparameter und Sonderfälle.

Diese Methodik deckt sich auch mit der im Abschnitt 5.1 dargelegten generellen Vorgehensweise zur Generierung von Handlungsalternativen.

Bevor im Folgenden auf die Details eines allgemeinen Algorithmus eingegangen wird, wird zunächst die Darstellung von Handlungsalternativen beleuchtet. Danach befassen sich zwei Abschnitte mit den bereits angesprochenen qualitativen, zeitlichen und quantitativen Steuerparametern, sowie Sonderfällen.

5.3.2 Darstellung von Handlungsalternativen

Die Ausgangsdaten zur Initialisierung eines Simulationslaufes werden in dem SEWS-spezifischen XML-Schema verwaltet (vgl. Abschnitt 4.1.3). Zur Abbildung einer Handlungsalternative wird eine Kopie dieser Daten erzeugt und die Steuerparameter angepasst.

Die Ergebnisse der Belegungsplanung kann mit Hilfe einer veränderten Prozesssequenz dargestellt werden. Jedes Produkt ist einem bestimmten Produkttyp zugeordnet, welches eine bestimmte Prozesssequenz in einem Materialflusssystem durchläuft. Diese Prozesssequenz wird in dem XML-Schema als *ProcessSequence* bezeichnet und beinhaltet eine Beschreibung (*Description*) und eine eindeutige *ProcessSequenceID* (vgl. Abb. 5.5). Mit Hilfe des Elements `<ProductProcess>` wird die für einen Produkttyp individuelle Prozesssequenz festgelegt, die eine Liste von Prozessen darstellt, die zuvor in dem Prozesssystem des Kind-Elementes `<ProcessSystem>` festgelegt wurden.

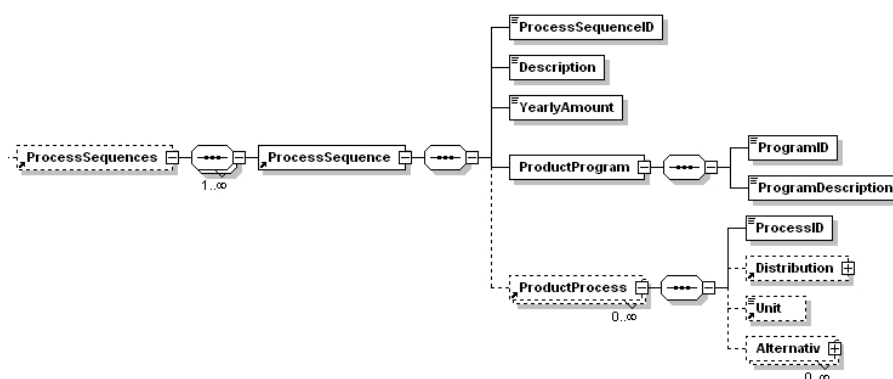


Abb. 5.5: Das Kind-Element *ProcessSequences* des XML-Schemas für SEWS

Darüber hinaus kann eine Jahresstückzahl hinterlegt werden, die bspw. zur Berechnung einer Taktzeit oder von Mengenverhältnissen herangezogen wird. Das Element `<ProductProgram>` dient zur Beschreibung der Art des Produktionsprogramms (Anm.: Auch hier über eine *Description* und eine eindeutige *ProductProgramID*) wie *Losgrößenfertigung*, *Just-In-Sequence (JIS)* oder *Just-In-Time (JIT)*. Aus diesem Produktionsprogrammtyp ergeben sich Konsequenzen für die Steuerung des Materialflusssystems, die bei der Modellentwicklung berücksichtigt werden müssen.

Zeitliche Steuerparameter werden über eine veränderte Auftragsreihenfolge verarbeitet. Diese wird in Form eines Produktionsplanes für den Prognosezeitraum verwaltet, der mit Hilfe des Elementes `<WorkOrderList>` dargestellt wird. Dieses Element enthält eine Liste der Aufträge (Anm.: Ein Auftrag entspricht dem Element `<Order>`) unter der Angabe einer *ProductID*, einer *ProductSequenceID*, einer *Description* und eines *Timestamp*.

Die Reihenfolge der Aufträge in dieser Liste entspricht der geplanten Abarbeitung im Prognosezeitraum. Die Abbildung unterschiedlicher Produktionsprogramme muss aufgrund ihrer Komplexität im Simulator erfolgen. Dabei wird mit Hilfe des Zeitstempels der gewünschte Auftragsbeginn festgelegt. Diese Zusammenhänge sind in Abbildung 5.6 dargestellt.

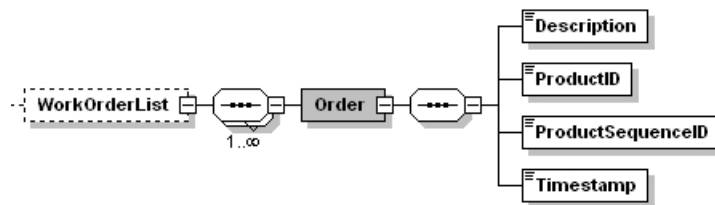


Abb. 5.6: Das Kind-Element *WorkOrderList* des XML-Schemas für SEWS

Auch die quantitativen Parameter sind bereits in dem globalen XML-Schema für SEWS berücksichtigt. Zur Kennzeichnung dieser veränderbaren Parameter für ein spezifisches Materialflusssystem, werden dem entsprechenden XML-Element Attribute beigefügt. Hierbei finden vier Attribute Anwendung:

- **alt_param**
Legt fest, ob dieses Element einen quantitativen Parameter darstellt. Ist dies der Fall, wird `alt_param = 'true'` gesetzt.
- **alt_lower**
Hier wird die untere Schranke des Wertebereiches definiert.
- **alt_upper**
Entspricht der oberen Schranke des Wertebereiches.
- **alt_step**
Legt die diskrete Schrittweite fest, die den Wertebereich von der unteren Schranke bis zur oberen Schranke aufteilt.

Eine Ausnahme bilden Sonderfälle, die über eine spezifische Alternativen-ID zu berücksichtigen sind. Diese müssen in dem Simulationsmodell erfasst und über diese eindeutige ID verwendet werden können.

Nach der Erläuterung der Darstellung von Handlungsalternativen im Zusammenhang mit der SEWS-internen Datenverwaltung, werden im folgenden Abschnitt die qualitative und zeitliche Belegungsplanung vorgestellt.

5.3.3 Qualitative und zeitliche Belegungsplanung

Im Allgemeinen ist unter der Belegungsplanung das klassische Scheduling zu verstehen, wobei ein Ablaufplan erstellt wird, der Prozessen zeitlich begrenzte Ressourcen zuweist ([Bru04]). Im Bereich von Produktion und Logistik wird dabei festgelegt, welche Aufträge wann und auf

welchen Produktionsanlagen ausgeführt werden sollen. Zur Klassifikation des Belegungsplanungsproblems sind sowohl Ressourcen- und Auftragscharakteristika als auch Optimalitätskriterien zu berücksichtigen.

Bei den Optimalitätskriterien kann es sich um durchlaufzeitbezogene, kapazitätsorientierte und terminorientierte Ziele handeln. Ressourcencharakteristika beschreiben sowohl die Ressourcenanzahl als auch die Ressourcenart und -anordnung. Reihenfolgebeziehungen, Prozesszeiten einzelner Arbeitsschritte und die Anzahl von Aufträgen werden in den Auftragscharakteristika berücksichtigt ([DSV97]).

Diese Klassifikationskriterien führen im Zusammenhang mit einem spezifischen Materialflusssystem zu Nebenbedingungen bzw. Restriktionen für den Belegungsplanungsalgorithmus. Darüber hinaus kommen verschiedene Lösungsalgorithmen in Betracht, die auf ein spezifisches Materialflusssystem angepasst werden müssen.

Um die Anzahl der notwendigen Simulationsläufe zu reduzieren, bietet sich die Festlegung der notwendigen Restriktionen und Optimalitätskriterien an. Hierzu müssen die zu detektierenden Ausnahmefälle auf die Restriktionen bzw. Nebenbedingungen gemappt werden. Dabei soll unter dem *Mapping* die Überführung bzw. Verknüpfung von Informationen in unterschiedlichen Darstellungsformen verstanden werden ([GB03]). Eine solche Einschränkung ist bereits bei der Definition von Ausnahmefällen vorzunehmen, da eine automatische Zuordnung nicht möglich ist. Hierzu müssten alle möglichen Kombinationen überprüft werden, was die Komplexität der Handlungsalternativengenerierung sehr stark erhöhen würde.

Die für SEWS definierten Ausnahmefälle werden hierbei über eindeutige IDs mit den Restriktionen und Optimalitätskriterien verknüpft. Die möglichen Restriktionen, die mit Ausnahmefällen im Bereich von Produktion und Logistik verbunden sind, sind im Folgenden:

- Die Beschränkung von Ressourcenverfügbarkeiten. Diese kann zeitlich befristet während des Prognosezeitraumes auftreten, oder den gesamten Prognosezeitraum überspannen.
- Verstärkte Kapazitätsbeschränkungen bspw. von Puffern, Ressourcen oder Werkern.
- Priorisierung von Aufträgen wegen Terminabweichungen.
- Optimalitätskriterien des Ausnahmefalls.

Dadurch wird im Voraus festgelegt, welche Restriktionen und Optimalitätskriterien in einem spezifischen Ausnahmefall herangezogen werden müssen. Tabelle 5.1 stellt ein beispielhaftes Mapping von Ausnahmefällen auf die Restriktionen der Belegungsplanung dar. Eine solche Tabelle soll im Folgenden als *Mappingmatrix* bezeichnet werden. Diese Vorgehensweise entspricht dabei nicht dem typischen Ansatz einer Alternativensammlung wie bei Monitoringsystemen, da die Handlungsalternativen selbst nicht vordefiniert sind.

Für jeden Ausnahmefall wird geprüft, ob mit ihm Restriktionen bzw. Optimalitätskriterien verbunden sind. Ist dies der Fall, wird eine Belegungsplanung mit den angepassten Restriktionen durchgeführt. Hierzu sind bestimmte Eingangsinformationen bzw. Kennwerte aus den Prognosewerten erforderlich. Diese werden mit Hilfe einer Kennwertdefinition ermittelt und sind mit

Mapping	Ausnahmefälle		
Restriktionen Belegungsplanung	Ausnahme 1: Puffer zu voll	Ausnahme 2: Aufträge zu spät	Ausnahme 3: Maschinenausfall
Ressourcenverfügbarkeit			X
Kapazitätsbeschränkungen	X		
Auftragspriorisierung		X	
Optimalitätskriterium			
Durchlaufzeit			X
Wartezeit			
Leerzeit			
Rüstzeit			
Terminabweichung	X		
Verspätung		X	

Tab. 5.1: Mapping von Ausnahmefällen auf die Restriktionen der Belegungsplanung

den entsprechenden Optimalitätskriterium über die Ausnahmefalldefinition im SEWS-internen XML-Schema und der Mappingmatrix verknüpft. Die Verwaltung dieser Mappingmatrix erfolgt mit Hilfe der Komponente Knowledge Base (vgl. Abschnitt 3.3.2).

Das Ergebnis ist eine Handlungsalternative in Form eines neuen Belegungsplanes, die sich durch einen veränderten Materialfluss auszeichnet. Dadurch kann die Anzahl durchzuführender Simulationsläufe bereits stark reduziert werden.

Neben der qualitativen und zeitlichen Belegungsplanung können zur Handlungsalternativen-generierung auch quantitative Steuerparameter und Sonderfälle herangezogen werden, womit sich der nachfolgende Abschnitt befasst.

5.3.4 Quantitative Steuerparameter und Sonderfälle

Im Bereich der quantitativen Steuerparameter und Sonderfälle ist ebenfalls eine effiziente und schnelle Bestimmung von Handlungsalternativen notwendig.

Hierzu eignet sich eine Zuordnungsmatrix, die einen Korrelationskoeffizienten als Zusammenhangsmass verwendet. Dieser beschreibt die Eignung eines Steuerparameters für einen spezifischen Ausnahmefall. Ein Zusammenhangsmass stellt im Allgemeinen die Stärke und die Art eines Zusammenhanges zweier Größen dar ([Bol04]).

Unter einem Korrelationskoeffizienten wird ein Zusammenhangsmass verstanden, welches Werte in dem Intervall $[-1; +1]$ annehmen kann. Der positive Maximalwert $+1$ weist darauf hin, dass ein sehr starker positiver Zusammenhang zwischen zwei Größen existiert, während der Wert -1 auf einen stark negativen Zusammenhang hinweist. Der Wert 0 deutet auf einen fehlenden Zusammenhang hin.

In SEWS wird eine *Koeffizientenmatrix* hinterlegt, in der die quantitativen Steuerparameter und die Sonderfälle mit den definierten Ausnahmefällen über eben einen solchen Koeffizienten verknüpft werden (vgl. Tab. 5.2).

Eine starke positive Korrelation deutet darauf hin, dass ausgehend vom gegenwärtigen Parameterwert eine Erhöhung des Steuerparameters in dessen Wertebereich erfolgsversprechend

Quantitative Einflussparameter / Sonderfälle	Ausnahmefälle			
	Ausnahme 1: Puffer zu voll	Ausnahme 2: Aufträge zu spät	Ausnahme 3: Maschinenausfall	...
<i>Werkertyp_1(i); i=2;...;4</i>	0,30	0,80	0,00	...
<i>Werkertyp_2(i); i=1;...;8</i>	0,50	0,90	0,30	...
<i>Schichtmodell(i); i=A,B,C</i>	0,00	0,70	0,50	...
<i>Pausenzeiten_1(i); i=10;20</i>	-0,80	-0,70	-0,50	...
<i>Ressource_1(i); i=1;2</i>	0,01	0,01	0,70	...
<i>Ressource_2(i); i=1;2</i>	0,01	0,01	1,00	...
<i>Sonderfall_1</i>	-1,00	0,00	0,00	...
<i>Sonderfall_2</i>	0,00	1,00	0,00	...
...

Tab. 5.2: Korrelationskoeffizienten für Ausnahmefälle und Steuerparameter

ist. Eine negative Korrelation bedeutet, dass eine negative Veränderung eine sinnvolle Handlungsalternative ergeben kann. Erst diese Korrelationsmatrix ermöglicht die schnelle Auswahl vielversprechender Steuerparameterveränderungen, die von der Komponente Knowledge Base verwaltet wird.

Es stellt sich nun die Frage, wie diese Matrix erstellt werden kann. Beim erstmaligen Einsatz eines SEWS für ein spezifisches Materialflusssystem werden die möglichen Steuerparameter und die Typen von Ausnahmefällen definiert. Es fehlen allerdings noch die Aussagen über den Korrelationskoeffizienten. Hierauf wird im Abschnitt 5.3.6 eingegangen. Zunächst erfolgt die Darstellung eines allgemeinen Algorithmus zur Handlungsalternativengenerierung.

5.3.5 Ein allgemeiner Algorithmus zur Handlungsalternativengenerierung

Den Ausgangspunkt des Algorithmus zur Handlungsalternativengenerierung in SEWS stellen die in den vorangehenden Abschnitten vorgestellte Koeffizienten- und Mappingmatrix dar.

Anhand der Mappingmatrix werden die Restriktionen zur Belegungsplanung abgeleitet. Sind keine Restriktionen vorhanden, so existiert für diesen Ausnahmefall kein Belegungsplanungsalgorithmus. Ansonsten wird eine Belegungsplanung durchgeführt, deren Ergebnis in den Steuerparametern dargestellt (vgl. Abschnitt 5.3.2) und mit der Simulation bewertet wird.

Mit Hilfe der Koeffizientenmatrix können quantitative Steuerparameter und Sonderfälle identifiziert werden, die zur Vermeidung eines Ausnahmefalls geeignet erscheinen. Dabei wird in diesem Abschnitt zunächst davon ausgegangen, dass die Koeffizientenmatrix bereits auf ein Materialflusssystem in geeigneter Weise adaptiert wurde (vgl. Abschnitt 5.3.4).

Zur Erläuterung des allgemeinen Algorithmus sind Symbole und Abkürzungen einzuführen, die eine geeignete Darstellung ermöglichen. Folgende Indizes werden verwendet:

- n : Repräsentiert die Anzahl der bei der Prognose detektierten Ausnahmefälle
- m : Stellt die Anzahl der Alternativenklassen in der Koeffizientenmatrix dar
- i, k, q : Laufindices für die detektierten Ausnahmefälle, mit $i, k, q = 1, \dots, n$
- j : Laufindex für die Alternativenklassen in der Koeffizientenmatrix, mit $j = 1, \dots, m$

Weiterhin sind Symbole für Mengen, Listen und Datensets notwendig, die zur Verwaltung und Darstellung von Informationen und Daten benötigt werden. Hierbei handelt es sich um

- Ω_{HA} : Die Menge der bei der Prognose detektierten Ausnahmefälle
- HA : Datenset der SEWS- und simulationsrelevanten Daten der Ausgangssituation für die ursprüngliche Prognose
- H_M : Symbol für die Mappingmatrix
- H_K : Symbol für die Koeffizientenmatrix
- S_{HA} : Liste der Alternativenklassen in der Koeffizientenmatrix
- Q_i : Menge der Kriterien in der Mappingmatrix für den Ausnahmefall i
- Θ_{AK} : Repräsentiert die Menge der Abbruchkriterien für eine bestimmte Schleife des Algorithmus.

Über Symbole für Listen, Mengen und Datensets hinaus sind lokale und globale Speichervariablen erforderlich, die eine Darstellung bzw. Repräsentation von Werten und Berechnungsergebnissen erlauben. Diese werden in der folgenden Liste erläutert.

- l_i : Stellt den definierten Grenzwert eines Ausnahmefalls i dar
- p_i : Repräsentiert den Prognosewert eines mit einem Ausnahmefall i verbundenen Kennwertes
- Δ_i : Variable zur Darstellung der Grenzwertüberschreitung, mit $\Delta_i = p_i - l_i$
- α_i : Variable zur Gewichtung der Grenzwertüberschreitung Δ_i , mit $\alpha_i \in [0; 1]$
- $c_{i,j}$: Korrelationskoeffizient der Koeffizientenmatrix einer Alternative j für den Ausnahmefall i
- $b_{i,j}$: Betrag des Korrelationskoeffizienten $c_{i,j}$, mit $b_{i,j} = |c_{i,j}|$
- $Min_{\Delta,i}$: Globale Speichervariable für die bisher geringste gefundene Grenzwertüberschreitung bezüglich des Ausnahmefalls i
- $Curr_{\Delta}$: Lokale Speichervariable für die Grenzwertüberschreitung
- $Min_{HA,i}$: Globale Speichervariable für den Datenset der SEWS- und simulationsrelevanten Daten, welche alle Handlungsalternativen enthält, die zur bisher geringsten Grenzwertüberschreitung des Ausnahmefalls i geführt haben
- $Curr_{HA}$: Lokale Speichervariable für einen Datenset der SEWS- und simulationsrelevanten Daten
- $\bar{p}_{i,Curr_{HA}}$: Neuer Prognosewert der mit einem Ausnahmefall i verbundenen Berechnungsregel, der durch Simulationsläufen mit dem Datenset $Curr_{HA}$ ermittelt wird

Zum Abschluss der Auflistung von Symbolen werden Funktionen aufgeführt, die Werte und Ergebnisse sowie Aktionen repräsentieren, die im Verlauf des Algorithmus notwendig sind und die Darstellung vereinfachen:

- f_Z : Repräsentiert die Zielfunktion der HA-Generierung
- $F_{calc}(f_Z)$: Berechnung des Ergebnisses der Zielfunktion auf Basis des Datensets der aktuellen Lösung
- $h(H_M, i)$: Funktion zur Bestimmung der Anzahl der Kriterien in der Mappingmatrix H_M bzgl. Ausnahmefall i

- φ_{i,Q_i} : Diese Funktion stellt die Durchführung der Belegungsplanung für den Ausnahmefall i dar
- $g_{HA}(j)$: Funktion zur Ermittlung neuer Eingangsdaten für die Simulation nach Anwendung der Alternative j
- $\Psi_{AK}(\Theta_{AK})$: Funktion, die untersucht, ob ein Abbruchkriterium aus der Menge Θ_{AK} erfüllt ist
- $f_{\alpha}(Curr_{\Delta}, Min_{\Delta,i}, i)$: Funktion zur Anpassung der Koeffizienten in der Koeffizientenmatrix in Abhängigkeit der Wirksamkeit bzgl. Ausnahmefall i

Nach der Erläuterung der einzelnen Indizes, Variablen, Mengen bzw. Datensets und Funktionen zur Darstellung des allgemeinen Algorithmus, wird dessen Funktionsweise zur Handlungsalternativengenerierung unter Verwendung der vorgestellten Symbole dargestellt.

Die Zielfunktion entspricht der Minimierung der Summe aller Grenzwertüberschreitungen Δ_i aller Ausnahmefälle i . Dabei wird die ausnahmefallspezifische Grenzwertüberschreitung mit einem Skalierungsfaktor α_i verbunden, der eine Gewichtung der einzelnen Ausnahmefälle ermöglicht. Alle Grenzwertüberschreitungen werden in der Menge Ω_{HA} gespeichert, die Ausnahmefälle nach ihrer Dringlichkeit bzw. dem größten $(\Delta_i \cdot \alpha_i)$ absteigend sortiert. Ziel des Algorithmus ist es, die Grenzwertüberschreitungen so gering wie möglich zu gestalten.

In einem zweiten Schritt versucht der Algorithmus für jeden Ausnahmefall i in der Menge Ω_{HA} nacheinander eine Belegungsplanung φ_{i,Q_i} durchzuführen, deren Ergebnis mit Hilfe der Simulation bewertet wird. Trägt diese zu einer Verbesserung oder gar Vermeidung des Ausnahmefalls i bei, wird die Handlungsalternative im Datenset $Min_{HA,i}$ gespeichert und für weitere direkt nachfolgende Belegungsplanungen in den Eingangsdaten mit einbezogen.

Die Überprüfung der Wirksamkeit der quantitativen Steuerparameter und Sonderfälle erfolgt wiederum für jeden Ausnahmefall $i \in \Omega_{HA}$. Hierfür wird eine Alternativenliste $S_{HA,i}$ erstellt, die alle Alternativenklassen j aus der Koeffizientenmatrix H_K enthält, für deren Korrelationskoeffizienten $c_{i,j} > 0$ gilt. Diese Liste wird dabei absteigend nach dem Betrag $b_{i,j}$ der Korrelationskoeffizienten sortiert.

Anschließend durchläuft der Algorithmus eine Schleife, solange der Ausnahmefall i noch nicht vermieden wurde und ansonsten kein Abbruchkriterium aus der Menge Θ_{AK} erfüllt ist. In dieser Schleife wird der Steuerparameter der ersten Alternativenklasse j aus der Alternativenliste $S_{HA,i}$ betrachtet und dessen Wert entsprechend der Angaben in der Koeffizientenmatrix erhöht oder verringert. Der Effekt dieser Veränderung auf den Ausnahmefall i wird mit einem Simulationslauf bewertet. Dies geschieht so lange, bis keine Verbesserung der Grenzwertüberschreitung mehr erreicht werden kann, dadurch kein neuer Ausnahmefall k , mit $k \notin i$ erzeugt wird oder die anderen Δ_q mit $q \neq i$ eine Verschlechterung erfahren.

Die gegenwärtig größte Verbesserung der Grenzwertüberschreitung $Min_{\Delta,i}$ des Ausnahmefalls i wird in einem globalen Datenset $Min_{HA,i}$ verwaltet, der alle bisher ermittelten Handlungsalternativen enthält. Der Algorithmus endet, sobald keine Alternativenklassen quantitativer Steuerparameter und Sonderfälle in der Alternativenliste $S_{HA,i}$ mehr betrachtet werden können. Dabei handelt es sich um ein Abbruchkriterium und es gilt entsprechend $\Psi_{AK}(\Theta_{AK}) = \text{'true'}$.

Der gesamte Algorithmus wird in Abbildung 5.7 übersichtlich dargestellt.

Allgemeiner Algorithmus bei Auftreten von n Ausnahmefällen

Bestimme die Zielfunktion f_Z

$$f_Z = \min \sum_{i=1}^n \Delta_i \cdot \alpha_i, \text{ mit } \Delta_i = p_i - l_i \text{ und } \alpha_i \in [0; 1]$$

Füge alle i in die Liste Ω_{HA} ein und sortiere diese absteigend nach $\Delta_i \cdot \alpha_i$

Für alle $i \in \Omega_{HA}$

Setze $Min_{\Delta,i} := \Delta_i$ und $Min_{HA,i} := HA$

Setze $Curr_{\Delta} := 0$ und $Curr_{HA} := null$

Falls $h(H_M, i) > 0$ und $F_{calc}(f_Z) > 0$ gilt

Durchführung einer Belegungsplanung Φ_{i,Q_i}

Durchführung von Simulationsläufen mit den Eingangsdaten $Curr_{HA} = g_{HA}(\Phi_{i,Q_i})$

Setze $Curr_{\Delta} = \bar{p}_{i,Curr_{HA}} - l_i$

Falls $Curr_{\Delta} < Min_{\Delta,i}$ und $\bar{p}_{q,Curr_{HA}} - l_q \leq Min_{\Delta,q}, \forall q \neq i$ gilt

Setze $Min_{\Delta,i} = Curr_{\Delta}$

Setze $Min_{HA,i} = Curr_{HA}$

Für alle $i \in \Omega_{HA}$

Erstellen einer Alternativenliste $S_{HA,i}$

Füge jede Alternative j in $S_{HA,i}$ ein, für die $c_{i,j} \neq 0$ gilt

Sortiere die Alternativen in $S_{HA,i}$ absteigend nach $b_{i,j}$

Solange $Min_{\Delta,i} > 0$, $F_{calc}(f_Z) > 0$ und $\Psi_{AK}(\Theta_{AK}) = \text{'false'}$ gilt

Wähle die erste Alternativenklasse $j \in S_{HA,i}$ aus und entferne diese aus $S_{HA,i}$

Solange keine Verbesserung von $Min_{\Delta,i}$ mehr erreichbar ist, keine neuen Ausnahmefälle k , mit $k \notin i$ auftreten und sich keine Δ_q verschlechtern, mit $q \neq i$

Veränderung der Eingangsparameter entsprechend der Koeffizientenmatrix, mit $Curr_{HA} = g_{HA}(j)$

Durchführung von Simulationsläufen mit den Eingangsdaten $Curr_{HA}$

Setze $Curr_{\Delta} = \bar{p}_{i,Curr_{HA,i}} - l_i$

Falls $Curr_{\Delta} < Min_{\Delta,i}$ und $\bar{p}_{q,Curr_{HA}} - l_q \leq Min_{\Delta,q}, \forall q \neq i$ gilt

Setze $Min_{\Delta,i} = Curr_{\Delta}$

Setze $Min_{HA,i} = Curr_{HA}$

Veränderung der Koeffizienten mit $f_{\alpha}(Curr_{\Delta}, Curr_{HA})$

Falls $S_{HA} = \emptyset$, gilt $\Psi_{AK}(\Theta_{AK}) = \text{'true'}$

Abb. 5.7: Allgemeiner Algorithmus zur Handlungsalternativengenerierung in SEWS

Im Folgenden wird auf die Anpassung bzw. Adaption der Korrelationskoeffizienten an ein spezifisches Materialflusssystem eingegangen.

5.3.6 Adaption der Korrelationskoeffizienten an ein Materialflusssystem

Im vorangehenden Abschnitt wird ein allgemeiner Algorithmus zur Generierung von Handlungsalternativen vorgestellt. Hierbei spielen die in Abschnitt 5.3.4 erläuterten Quantitativen Steuerparameter und Sonderfälle eine große Rolle, deren Verknüpfung mit möglichen Ausnahmefällen über so genannte Korrelationskoeffizienten erfolgt.

Diese Koeffizienten bewegen sich in dem Intervall $[-1; 1]$ und repräsentieren die *Eignung* einer Handlungsalternative zur Verbesserung oder Vermeidung eines Ausnahmefalls. Dabei wurde bisher zunächst angenommen, dass die Koeffizienten bereits in geeigneter Weise hinterlegt bzw. angepasst worden sind.

In diesem Abschnitt wird geklärt, wie und wann eine geeignete Anpassung der Koeffizienten erfolgen kann. Dabei können unterschiedliche Phasen unterschieden werden:

- **Erstmalige Implementierung**

Bei der erstmaligen Implementierung kann die Geschwindigkeit des Adaptionsverfahrens und somit die gesamte Performance des SEWS durch die manuelle Angabe realistischer Korrelationskoeffizienten durch den Anwender erhöht werden. Dadurch ist es bereits im Vorfeld möglich, unwirksame Steuerparameter im Falle einer Ausnahme auszuschließen.

- **Produktiveinsatz**

Während des realen Betriebes generiert das SEWS im Ausnahmefall Handlungsalternativen, die mit der Simulation bewertet werden. Hierbei findet eine Rückkopplung auf den Korrelationskoeffizienten zwischen Steuerparameter und Ausnahmefall statt. In dieser Phase erfolgt die Anpassung während der Durchführung des Allgemeinen Algorithmus zur Handlungsalternativengenerierung und wird durch die Funktion $f_{\alpha}(Curr_{\Delta,i}, Min_{\Delta,i}, i)$ repräsentiert (vgl. Abb. 5.7).

- **Parallel zur produktiven Nutzung**

Parallel während des realen Einsatzes eines SEWS sind auf Basis von Realdaten die Korrelationskoeffizienten veränderbar. Hierzu werden die mit den Ausnahmefällen verknüpften Berechnungsregeln verwendet und die Auswirkungen der einzelnen Steuerparameter ermittelt. Das Simulationsmodell erzeugt also Ergebnisse für reale Basisdaten, wofür die Steuerparameter variiert und deren Effekt auf die Berechnungsregeln der Ausnahmefälle berücksichtigt werden. Hierzu ist ein entsprechender *Trainingsplan* zu erstellen, wobei dieses *Training* sowohl manuell als auch automatisiert erfolgen kann.

Es ist eine Trainingsfunktion einzuführen, die bei der Veränderung der Korrelationskoeffizienten sowohl die Anzahl der Trainingszyklen als auch die Stärke der Beeinflussung durch einen Steuerparameter berücksichtigt.

Wird ein Ausnahmefall i definiert, so sind die geeigneten Alternativen zunächst unbekannt und die Korrelationskoeffizienten können mit einem willkürlichen Startwert belegt werden. Im Rahmen dieser Arbeit wird für jede Alternative j der Korrelationskoeffizient auf $c_{i,j} = 0.5$ bzw. $c_{i,j} = -0.5$ gesetzt. Hierfür kann angenommen werden, dass zumindest der Zusammenhang der Richtung des Koeffizienten zu Beginn bekannt ist.

Um eine Anpassung durchzuführen, muss eine Variable $\pm\Delta_{i,j}$ eingeführt werden, um die der Korrelationskoeffizient $c_{i,j}$ in Abhängigkeit der Beeinflussungsstärke des Ausnahmefalls i durch die Alternative j verändert wird. Diese Anpassung während des Produktiveinsatzes geschieht mit Hilfe der Funktion $f_\alpha(Curr_\Delta, Min_{\Delta,i}, i)$ bei der Durchführung des in Abbildung 5.7 dargestellten Algorithmus.

Eine einfache Möglichkeit ist es, den Startwert für $\Delta_{i,j}$ in Abhängigkeit des Startwertes des Korrelationskoeffizienten $c_{i,j}$ zu wählen, mit $\Delta_{i,j} = c_{i,j}/2$. Mit jedem Trainingszyklus wird der Wert dieser Variablen halbiert, wodurch der Einfluss mit zunehmender Anzahl an Lernzyklen abnimmt. Die Veränderung des Korrelationskoeffizienten erfolgt in Abhängigkeit des Ergebnisses der Beeinflussung des Ausnahmefalls durch die Alternative. Hierbei gibt es drei grundsätzliche Möglichkeiten:

1. Positive Beeinflussung

Für das Ergebnis der Verbesserung der Grenzwertüberschreitung gilt $Min_{\Delta,i} - Curr_\Delta > 0$. Der Korrelationskoeffizient kann also entsprechend positiv verändert werden, mit $c_{i,j}^{neu} = c_{i,j}^{alt} + \Delta_{i,j}$ und $\Delta_{i,j} = c_{i,j}^{neu}/2$.

2. Negative Beeinflussung

Mit $Min_{\Delta,i} - Curr_\Delta < 0$ liegt eine negatives Ergebnis für die Verbesserung der Grenzwertüberschreitung vor. Deshalb gilt für die Veränderung des Korrelationskoeffizienten der Zusammenhang $c_{i,j}^{neu} = c_{i,j}^{alt} - \Delta_{i,j}$ und $\Delta_{i,j} = c_{i,j}^{neu}/2$.

3. Neutralität

Das Ergebnis der Verbesserung führt zu keiner Veränderung, mit $Min_{\Delta,i} - Curr_\Delta = 0$. Hier wird keine Anpassung vorgenommen.

Im Falle einer mehrmaligen Neutralität eines Ausnahmefalls i durch die Alternative j , ist die generelle Einflussmöglichkeit anzuzweifeln und der Koeffizient auf 0 zu setzen, wodurch die Anzahl durchzuführender Simulationsläufe reduziert wird.

Anschließend wird der Wert der Variablen so lange verringert, bis er einen bestimmten Wert unterschreitet. An dieser Stelle soll hierfür der Grenzwert ± 0.01 gewählt werden, unter den $\Delta_{i,j}$ niemals fallen darf. Dies garantiert die weitere Veränderbarkeit des Koeffizienten im weiteren Verlauf und es verhindert das dauerhafte *Festfahren* auf einem ungerechtfertigten Wert.

Falls das erste Mal der Korrelationskoeffizient nach einer Anpassung auf 0 gesetzt werden würde, wird statt dessen $c_{i,j}^{neu} = \text{sgn}(c_{i,j}^{alt}) \cdot (-1) \cdot 0.01$ gewählt. Hierdurch soll die eventuelle Eignung der entgegengesetzten Richtung getestet werden, um auszuschließen, dass bei der Ausnahmefalldefinition nicht das falsche Vorzeichen gewählt wurde.

Die Anpassung der Korrelationskoeffizienten ist sehr bedeutsam für die Performance eines SEWS, da dadurch ungeeignete Alternativen nicht bzw. immer seltener zur Handlungsalternativengenerierung für einen bestimmten Ausnahmefall herangezogen werden. Je schneller die Anpassung erfolgt, desto schneller ist das SEWS praktisch einsetzbar. Eine geeignete Vorgehensweise besteht in der Durchführung eines *Trainings* parallel zur produktiven Nutzung auf Basis von Historiedaten.

Um eine ungeeignete Alternative j für einen Ausnahmefall i in der Koeffizientenmatrix auf 0 zu setzen, sind bei der hier vorgestellten Vorgehensweise mit einem Startwert von $c_{i,j} = 0.5$ nur 10 Bewertungen erforderlich. In Abhängigkeit der Anzahl n zu detektierender Ausnahmefälle beträgt also die Anzahl der durchzuführenden Trainingsläufe bei paralleler Anpassung während der produktiven Nutzung $10 \cdot n$ Simulationsläufe. Anschließend stellt die Koeffizientenmatrix eine bereits sehr gute Näherung an ein spezifisches Materialflusssystem dar. Werden neue Typen von Ausnahmefällen definiert, müssen nur für diesen Ausnahmefall Bewertungen erfolgen.

Ein Nachteil entsteht, wenn bei der Ausnahmefalldefinition das falsche Vorzeichen gewählt wird. Dann erhöht sich diese Anzahl und es müssen mehr Simulationsläufe durchgeführt werden.

5.3.7 Einführung von Abbruchkriterien

Der in Abschnitt 5.3.5 dargestellte allgemeine Algorithmus zur Handlungsalternativengenerierung verwendet mehrere Schleifen, in denen die Steuerparameter des realen Materialflusssystems bzw. die Eingangsdaten für die simulative Bewertung verändert und anschließend Simulationsläufe durchgeführt werden.

Da dies sehr rechenaufwändige Vorgänge und Prozesse sind, müssen Abbruchkriterien eingeführt werden, die eine Beendigung des Algorithmus garantieren und die implizit die Performance eines SEWS berücksichtigen. Grundsätzlich sind beliebige Abbruchkriterien denkbar, die an die Ergebnisse der Simulationsläufe, die aufgetretenen Ausnahmefälle, die Ergebnisse der Berechnungsregeln bzw. Kennwerte oder die Anzahl der bereits durchgeführten Berechnungsschritte geknüpft sind.

Im Rahmen des hier vorgestellten Algorithmus beschränken sich die Abbruchkriterien auf die folgenden drei Möglichkeiten, die in der Menge Θ_{AK} verwaltet werden. Dabei wird der Algorithmus abgebrochen, wenn

- keine Verbesserung der Grenzwertüberschreitung $Min_{\Delta,i}$ eines Ausnahmefalls i mehr möglich ist,
- keine Alternativen mehr existieren, die zu einer Verbesserung der Grenzwertüberschreitung führen könnten und
- die Anzahl der zu überprüfenden Handlungsalternativen bzw. Simulationsläufe einen bestimmten Grenzwert überschreiten.

Die ersten beiden Abbruchkriterien werden im vorgestellten Algorithmus bereits verwendet, während das dritte Abbruchkriterium näher erläutert werden muss.

Wenn das Ergebnis der Handlungsalternativengenerierung noch vor dem nächsten Prognosezyklus verfügbar sein soll, so ist die maximal mögliche Anzahl durchführbarer Simulationsläufe c_{Sim} abhängig von der festgelegten Zeitspanne Δt_{Det} zwischen den Detektionszeitpunkten (vgl. Abschnitt 4.3.2.2). In Abhängigkeit der durchschnittlichen Dauer eines Simulationslaufes \bar{t}_{Sim} , ergibt sich c_{Sim} ohne parallele Durchführung von Simulationsläufen zu

$$c_{Sim} = \frac{\Delta t_{Det}}{\bar{t}_{Sim}}.$$

Eine bessere Performance der verwendeten Rechnersysteme oder ein verstärkter Einsatz der verteilten parallelen Simulation erhöht die Anzahl durchführbarer Bewertungen.

Mit Hilfe der Parallelisierung soll eine Beschleunigung von Simulationsexperimenten erreicht werden ([Fuj00]). Dies kann zum einen dadurch geschehen, dass voneinander unabhängige Simulationsläufe gleichzeitig mit verschiedenen Parameterkonstellationen auf verschiedenen Prozessoren ausgeführt werden. Hängen die Simulationsläufe voneinander ab, so ist dieser Ansatz ungeeignet. Zum anderen kann eine Parallelisierung durch Auslagerung bestimmter Hilfsfunktionen wie bspw. die Ereignisverwaltung oder die Generierung von Zufallszahlen auf andere Prozessoren erreicht werden.

Vor allem zu Beginn eines SEWS-Einsatzes für ein spezifisches Materialflusssystem sind mehr Simulationsläufe notwendig, da zunächst die Korrelationskoeffizienten entsprechend zu adaptieren sind. Hierbei ist es sinnvoll die maximale Anzahl möglicher Simulationsläufe in Abhängigkeit von c_{Sim} als Abbruchkriterium zu verwenden.

5.4 Zusammenfassung des Kapitels

Dieses Kapitel befasst sich mit der Generierung von Handlungsalternativen im Ausnahmefall, wozu in einem ersten Abschnitt die hierfür generelle Vorgehensweise vorgestellt wird. Die Schwerpunkte hierbei sind der Ablauf der Handlungsalternativengenerierung, die generellen Steuerparameter eines realen Materialflusssystems und die Begründung der Notwendigkeit eines Lösungsverfahrens.

Ein weiterer Abschnitt diskutiert die Einordnung und Abgrenzung der Problemstellung der Handlungsalternativengenerierung. Hierzu werden die Technologien Fuzzy-Logik, Künstliche Neuronale Netze, Regelbasierte Systeme und Expertensysteme sowie die Thematik der Simulation und Optimierung aufgegriffen.

Ein abschließender Abschnitt stellt ein Verfahren zur Ermittlung und Bewertung von Handlungsalternativen für Simulationsbasierte Frühwarnsysteme vor. Dabei wird eine mehrstufige Vorgehensweise begründet, die Darstellung von Handlungsalternativen in der SEWS-internen Datenverwaltung erläutert und sowohl auf die zeitliche und qualitative Belegungsplanung als

auch auf quantitative Steuerparameter und Sonderfälle eingegangen. Weiterhin erfolgt die Vorstellung eines allgemeinen Algorithmus zur Handlungsalternativengenerierung und die Adaption der Korrelationskoeffizienten an ein Materialflusssystem. Zum Abschluss werden Abbruchkriterien eingeführt, die eine Beendigung des Algorithmus garantieren.

Die in diesem Kapitel aufgezeigte Methodik berücksichtigt sowohl eine Klassifizierung möglicher Ausnahmefälle zur Verringerung der Anzahl zu betrachtender Alternativen als auch eine adaptive Vorgehensweise zur Anpassung eines SEWS an ein spezifisches Materialflusssystem. Die verschiedenen Steuerparameter eines Materialflusssystems werden anhand ihrer Beeinflussungsmöglichkeit bezüglich bestimmter Ausnahmefälle ausgewählt und nach der Bewertung mit Hilfe der Simulation eingesetzt. Dabei kommen zunächst diejenigen Steuerparameter zur Anwendung, die entsprechend des Wertes ihres Korrelationskoeffizienten die größte Erfolgswahrscheinlichkeit versprechen.

Dabei ermöglicht die mehrstufige Vorgehensweise eine Kombination verschiedener Handlungsalternativen, indem eine bisher erreichte Verbesserung mit ihren entsprechendem Datenset (vgl. Abschnitt 5.3.5) als Eingangswerte für die Überprüfung der nächsten verwendbaren Handlungsalternative bzw. Steuerparameter berücksichtigt wird.

Zur Handlungsalternativengenerierung müssen in Abhängigkeit eines spezifischen Materialflusssystems und den zu detektierenden Ausnahmefällen entsprechende Belegungsplanungsalgorithmen bzw. -verfahren entwickelt werden. Dabei sind die zu berücksichtigenden Parameter vollständig in den SEWS- und simulationsrelevanten Daten enthalten, die als Eingangsdaten für die Simulation dienen.

Eine durchdachte Ausnahmefalldefinition und eine geeignete Anpassung der Korrelationskoeffizienten entsprechend der in Abschnitt 5.3.6 erläuterten Vorgehensweise ermöglicht eine schnellere Generierung, da bereits frühzeitig ungeeignete Handlungsalternativen vernachlässigt werden können.

Die Handlungsalternativengenerierung ist weiterhin ein komplexes Themengebiet, welches noch für die nächsten Jahre die Entwicklung eines standardisierten Softwarewerkzeuges für alle möglichen Materialflusssysteme zur reinen Utopie macht. Dennoch ist bei entsprechender Datenpflege sowie Anpassung und Optimierung der Verfahren und Algorithmen eine Verbesserung der Ergebnisse möglich. Der im Rahmen dieser Arbeit vorgeschlagene Algorithmus ist zu verfeinern, damit schnellere und adäquatere Handlungsalternativen geliefert werden können. An dieser Stelle kann es sich nur um einen ersten vorsichtigen Vorschlag handeln. Dabei ist die Idee, die Anzahl möglicher Handlungsalternativen durch die Kombination einer Klassifizierung und einem Adaptionsverfahren zu reduzieren aus heutiger Sicht ein sinnvoller Ansatz. Darüber hinaus ist die Konfiguration des Adaptionsverfahrens weiter zu verbessern.

Kapitel 6

Prototypische Anwendung Simulationsbasierter Frühwarnsysteme

Der Unterschied zwischen Theorie und Praxis ist in der Praxis weit höher als in der Theorie.

— Ernst Ferstl

In diesem Kapitel werden die bisher vorgestellten theoretischen Ansätze im Rahmen Simulationsbasierter Frühwarnsysteme hinsichtlich ihrer praktischen Umsetzbarkeit untersucht. Hierzu befasst sich ein erster Abschnitt mit der Bewertung der technischen Umsetzung im betrieblichen Alltag eines Industrieunternehmens im Allgemeinen bzw. eines Automobilkonzerns im Besonderen. Ein weiterer Abschnitt geht auf repräsentative Fallbeispiele ein. Zu diesen Beispielen gehören die FSG-Montage und der Härteofen A28, die wichtige Teilprozesse in der Getriebeproduktion der DaimlerChrysler AG repräsentieren. Hierzu werden jeweils die Materialflusssysteme beschrieben, die entsprechenden Simulationsmodelle und die Datenversorgung diskutiert, die jeweilige Ausnahmedetektion und Handlungsalternativengenerierung vorgestellt und auf interessante Ergebnisse eingegangen.

6.1 Bewertung der technischen Umsetzung

In diesem Abschnitt wird die technische Umsetzung Simulationsbasierter Frühwarnsysteme in der betrieblichen Umgebung eines Industrieunternehmens betrachtet. Schwerpunkte dabei sind der Einsatz von Webtechnologien und Webservices zur Realisierung einer verteilten Architektur, die Datenbeschaffung und -versorgung, sowie die verwendeten Simulatoren.

6.1.1 Verteilte Systeme, Web Services und XML-Datenbanken

Der Einsatz von Web Services stellt bestimmte Anforderungen an die informationstechnische Infrastruktur eines Unternehmens. Es ist ein verteiltes System erforderlich, welches Komponenten wie Simulatoren, XML-Datenbanken, Webanwendungen und Web Services miteinander verknüpfen und verwalten kann. Hierzu sind leistungsfähige Rechner notwendig, auf denen Webserver zur Verwaltung der einzelnen Komponenten zu implementieren sind. Weiterhin müssen die verfügbaren Kapazitäten und Übertragungsraten eines Netzwerkes berücksichtigt werden, da sie starken Einfluss auf die Performance eines verteilten Systems nehmen.

XML-Datenbanken sind im Automobilbereich selten bis gar nicht vorzufinden, da XML im Vergleich zu historisch gewachsenen IT-Systemen eine sehr junge Technologie ist und vor allen Dingen im Bereich von Webanwendungen und zum Datenaustausch zwischen unterschiedlichen Systemen zur Anwendung kommt. Die meisten Anwendungen und Systeme basieren auf relationalen *Datenbankmanagementsystemen (DBMS)* wie *Oracle* oder *DB2*.

Im Verleich hierzu durchdringen Webanwendungen immer stärker den betrieblichen Alltag eines Industrieunternehmens im Allgemeinen und der Automobilindustrie im Besonderen. Vor allem Kunden- und Lieferantenbeziehungen werden über so genannte Internetportale gepflegt und aufgebaut ([GK05]). Verschiedene IT-Systeme werden mit einem zusätzlichen Web-Interface ausgestattet, um einen unternehmensweiten Zugriff zu erleichtern. Unter diesem Aspekt finden auch verstärkt Web Services Anwendung, wodurch die Einbindung von IT-Systemen in neue und übergeordnete Applikationen ermöglicht wird. Ein Beispiel hierfür sind so genannte Workflow-Systeme, die Geschäftsprozesse nachbilden und dem verantwortlichen Mitarbeiter zum richtigen Zeitpunkt die richtigen Informationen zur Verfügung stellen.

Die vollständige Koffiguration und Verwaltung eines SEWS erfolgt über eine Webanwendung. Abbildung 6.1 zeigt eine Webseite dieser Webanwendung, die als Benutzerschnittstelle die Aufgaben der Komponente User-Interface übernimmt (vgl. Abschnitt 3.3.2). Diese Benutzerschnittstelle kann jederzeit um weitere Funktionalitäten und Schnittstellen zu anderen IT-Systemen erweitert und als Intranetportal eingesetzt werden.

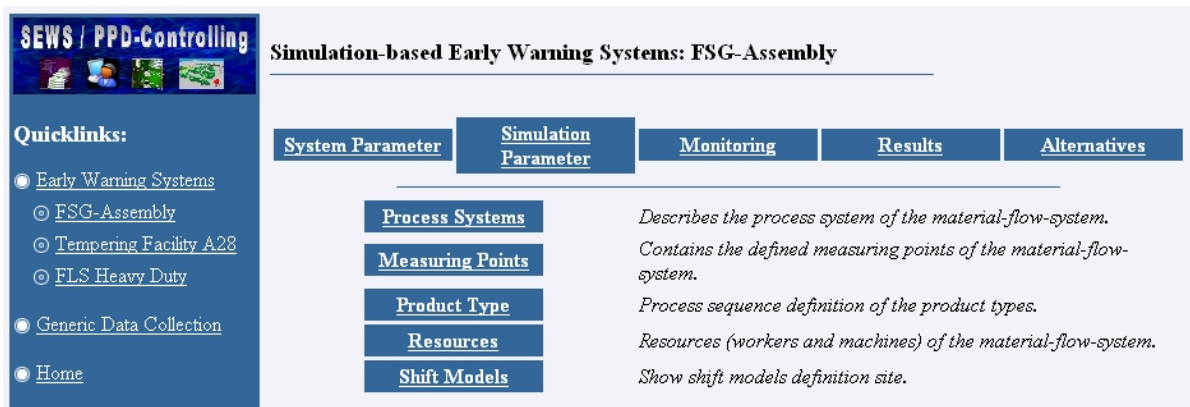


Abb. 6.1: Benutzerschnittstelle zur Verwaltung Simulationsbasierter Frühwarnsysteme

Die Kapselung von Simulatoren mit Hilfe eines Web Service stellt weitere Anforderungen an die hierfür notwendigen Softwarekomponenten auf einem Rechnersystem. Hierzu muss der Simulator und weiterhin ein Webserver auf diesem Rechner verfügbar sein, um dort den entsprechenden Web Service zu implementieren.

Dabei sind in den heutigen Industrieunternehmen leistungsfähige IT-Infrastrukturen zu finden, die ausreichend Rechenkapazitäten und performante Netzleitungen zur Verfügung stellen. Auch im Umgang mit Webdiensten, Webservern und Webanwendungen sind Erfahrungen, sowie entsprechende Hard- und Softwarekomponenten bereits vorhanden. Weiterhin ist eine ausreichende Sicherheit bei einem firmeninternen Einsatz verteilter Systeme gegeben, da die

Netzwerke nach außen hin stark gesichert sind und unternehmensintern jede Aktion der beteiligten Rechner überwacht werden kann. Diese technischen Rahmenbedingungen begünstigen den Einsatz und die Entwicklung verteilter Systeme, Web Services und XML-Datenbanken im Rahmen Simulationsbasierter Frühwarnsysteme.

Im Folgenden wird allgemein auf die Thematik der Datenbeschaffung und -versorgung eingegangen.

6.1.2 Datenbeschaffung und -versorgung im praktischen Umfeld

Die Datenbeschaffung und -versorgung im praktischen Umfeld spielt für die Umsetzbarkeit von SEWS eine große Rolle. Dabei sind vor allen Dingen die in Abschnitt 4.1.2 diskutierten Kriterien zur Systemauswahl relevant.

Sowohl die Performance als auch die Abfrage- und Anbindungskomplexität stellen in der Praxis keine wesentlichen Hindernisse dar. Die Informationssysteme verwenden im Hintergrund überwiegend moderne Datenbanken bzw. DBMS wie Oracle- oder DB2-Datenbanken, die eine einfache Integration in verschiedenste Applikationen ermöglichen. Dies unterstützt und vereinfacht die Datengewinnung aus originären Datenquellen, so dass die in Abschnitt 4.1.5 dargestellte Vorgehensweise zur Anwendung kommt.

Aus Produktivsystemen, unter denen die Fertigungs- und Montageleitstände verstanden werden, lassen sich alle prognoserelevanten Daten ersten Grades ermitteln (vgl. Abschnitt 4.1.1). Die an den Messpunkten erfassten Veränderungen im Zustand des Systems werden in den darunter liegenden Datenbanken in Sekundenbruchteilen angeglichen. Dies erleichtert die Aktualisierung von Modelldaten, da die zum Zeitpunkt der Datenbankabfrage gewonnenen Daten den entsprechenden Zustand des realen Systems repräsentieren. Eine Synchronisation mit anderen Systemen oder das Abwarten eines bestimmten Zeitpunktes oder -raumes zum Datenabzug ist nicht erforderlich.

Die statischen Daten bzw. prognoserelevanten Daten zweiten Grades finden sich in Informationssystemen, die über Batch-Läufe versorgt werden (vgl. Abschnitt 4.1.6). Der Sinn von Batch-Läufen liegt in der Sammlung von Datenänderungen und die konzentrierte Einspeisung in die zugrunde liegende Datenbank zu einem günstigeren Zeitpunkt als zu dem, an dem die Veränderungen tatsächlich vorgenommen wurden. Zielsetzung ist eine Verteilung der Rechen- und Netzlast in einer innerbetrieblichen IT-Infrastruktur. Aus diesem Grund werden Batch-Läufe entweder nächtlich oder jedes Wochenende durchgeführt. Dies ist abhängig davon, wie aktuell die veränderten Informationen in den entsprechenden Systemen sein müssen. Dadurch wird wiederum die Aktualisierung von Modelldaten erleichtert, da keine komplexen *Listener* benötigt werden, sondern eine zeitgesteuerte Aktualisierung erfolgen kann, die lediglich nach Beendigung eines Batch-Laufes vorzunehmen ist.

Problematisch sind im Allgemeinen die Datenqualität und -aktualität, die im menschlichen Verantwortungsbereich liegen. Vor allen Dingen bei einer manuellen Rückmeldung an Messpunkten kommt es vor, dass Daten zu spät, verfälscht oder gar nicht in die Informationssysteme eingepflegt werden.

Eine weitere Schwierigkeit im praktischen Umfeld stellen politische Rahmenbedingungen dar. Es kommt häufig vor, dass auf geeignete Systeme überhaupt nicht zugegriffen werden kann. Dabei kommen bspw. Betriebsvereinbarungen zum Tragen, die den Zugriff auf personenbezogene Daten beschränken. Aber auch organisatorische Hindernisse spielen hier eine Rolle, wenn zum Beispiel Daten über mehrere Datenbanken hinweg benötigt werden, die in unterschiedlichen Abteilungen zu finden sind.

Insgesamt stellte sich die Datenbeschaffung und -versorgung für die prototypische Umsetzung Simulationsbasierter Frühwarnsysteme aus praktischer Sicht als wenig problematisch heraus.

6.1.3 Verwendete Simulatoren

Zur Entwicklung von SEWS stehen im Rahmen dieser Arbeit zwei Simulationswerkzeuge zur Verfügung. Dabei handelt es sich um die Simulatoren *AutoMod* von dem Unternehmen *Brooks* ([Bro06]) und *SLX* von der Firma *Wolverine* ([Wol06]).

AutoMod ist eine grafik-orientierte Simulationsentwicklungsumgebung und nach dem Bausteinkonzept aufgebaut (vgl. Abschnitt 2.2.3.1). Über einen Modelleditor kann das Modell mit Hilfe eines Förder- und Prozess-Systems erstellt werden. Das Fördersystem enthält Bausteine zur Darstellung fahrerloser Transportsysteme, Stetigförderer, Einschienenhängebahnen, Hochregallager und Kräne. Ein Graphikmodul ermöglicht die Dimensionierung, Konfiguration und Visualisierung dieser Fördersysteme. Das Prozess-System übernimmt die Steuerung des gesamten Simulationsmodells, da hier unter anderem Fördergüter, Ressourcen, Auftragslisten, Variablen und Durchlaufpläne definiert werden können. Mit Hilfe der Durchlaufpläne wird für jedes Fördergut die Bearbeitungsreihenfolge und der Transportweg durch das Modell festgelegt. Zur Durchführung von Simulationsexperimenten muss das Modell kompiliert und zur Ausführung in das Simulationsmodul geladen werden ([Kud00]).

Das *SLX*-Simulationssystem gehört zur Gruppe der simulationsspezifischen Sprachen (vgl. Abschnitt 2.2.3.1), die aus der *SLX*-Sprache und Komponenten zur Entwicklung von Simulationssystemen besteht ([SH98]). Dabei ist *SLX* eine objektbasierte Sprache, die zum einen Anweisungen zur Simulation diskreter Prozesse und zum anderen allgemeine Anweisungen einer höheren Programmiersprache besitzt. Die simulationsspezifischen Anweisungen unterstützen die Modellierung diskreter Prozesse und werden durch die Simulationssprachen *GPSS/H* und *SIMULA* beeinflusst. Dagegen lehnen sich die allgemeinen Sprachbestandteile an *C/C++* an. Dabei ist unter *SLX* keine reine *C++* Klassenbibliothek zur Durchführung von Simulationen zu verstehen, da wesentlich mehr Funktionalitäten angeboten werden.

Mit beiden Simulatoren ist es möglich, komplexe Materialflusssysteme nachzubilden, wie sie im Bereich von Produktion und Logistik auftreten können.

6.2 Beschreibung der prototypischen Anwendungen

Zur Evaluation von SEWS wurden zwei prototypische Fallbeispiele entwickelt, die aus dem Bereich der Getriebeproduktion des Werkes Gaggenau und des Getriebewerkes Rastatt der DaimlerChrysler AG stammen.

Das Getriebewerk Rastatt stellt mit einer Jahresproduktion von ca. 286.000 Getrieben das Kompetenzzentrum für manuelle und automatisierte Schaltgetriebe der DaimlerChrysler AG dar und beliefert weltweit sowohl konzerninterne als auch -externe Kunden. Das Leistungsangebot reicht von Getrieben für PKW und Transporter bis hin zu Aggregaten für schwere LKW, Omnibusse und Spezialfahrzeuge. Zu den ausgewählten Bereichen für die Fallbeispiele gehören die Frontschaltgetriebemontage und der Härteofen A28. Im Folgenden werden die einzelnen Fallstudien vorgestellt.

6.2.1 Die Frontschaltgetriebemontage

Das Frontschaltgetriebe (FSG) ist ein Produkt des Getriebewerkes Rastatt der DaimlerChrysler AG und wird für die A-Klasse und B-Klasse der Marke Mercedes-Benz hergestellt. Dabei handelt es sich um ein voll synchronisiertes, mechanisches 6-Gang-Schaltgetriebe zum Einbau in ein frontgetriebenes Fahrzeug mit quer eingebautem Motor. Es existieren jeweils zwei unterschiedliche Varianten für Fahrzeuge mit Diesel- oder Benzinmotor, was sich in unterschiedlichen Dimensionierungen der entsprechenden Zahnräder und Antriebswellen niederschlägt.

6.2.1.1 Beschreibung des Materialflusssystemes

Das Montagesystem für das Frontschaltgetriebe besteht aus drei Montagekreisläufen mit insgesamt 35 Montagestationen, welche von jedem Getriebe durchlaufen werden (vgl. Abb. 6.2). In dem ersten Kreislauf erfolgt die Montage des so genannten Wellenpaketes, welches aus zwei Abtriebs- und einer Antriebswelle besteht, auf die Zahnräder aufgedrückt werden.

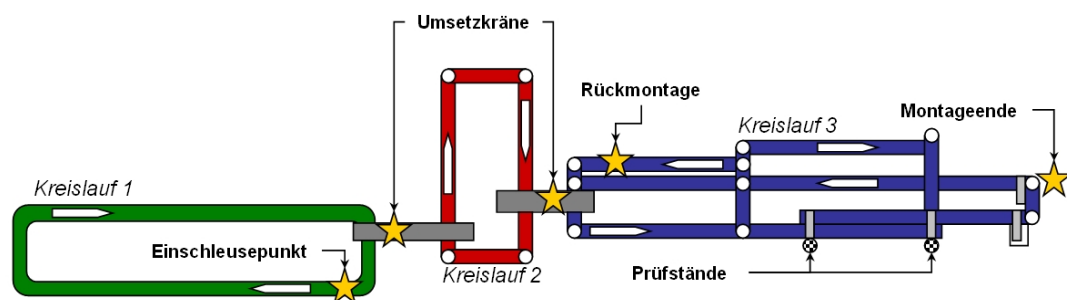


Abb. 6.2: Schematisches Layout der FSG-Montage

Die Montage des Differenzials, des Gehäusevorderteils, des Schaltungsdorns, sowie die Verschraubung des Gehäuses erfolgt im zweiten Kreislauf. Bis zum Ende dieses Kreislaufes begleitet ein Werker von der ersten Station an jeweils ein Getriebe, was die Produktverantwortung und auch die Motivation der Mitarbeiter steigert.

Der dritte Kreislauf ist voll automatisiert und dient zur Ölbefüllung und Überprüfung der Dichtigkeit des Getriebes, sowie der Prüfung auf die volle Funktionsfähigkeit in einem von insgesamt zwei identischen Prüfständen. Tritt hier ein Fehler auf, so wird das Getriebe in die Rückmontage befördert. Ist das Prüfergebnis positiv, so wird das Getriebe am Ende der Montage auf eine Palette abgesetzt und steht zum Abtransport bereit.

Das Materialflusssystem wird in zwei Schichten betrieben, welches eine effektive tägliche Arbeitszeit von 935 Minuten ergibt. Je nach Umfang der Montageaufträge arbeiten mindestens fünf, jedoch maximal acht Werker in der Montage, wodurch die Produktionskapazität von 280 auf 370 Getriebe pro Tag erhöht werden kann.

6.2.1.2 Simulationsmodell und Datenversorgung

Das Simulationsmodell zur Darstellung der FSG-Montage erfolgte mit Hilfe des Simulators AutoMod. Dabei wurde das Materialflusssystem anhand eines Layouts im DXF-Format entsprechend der realen Größenverhältnisse aufgebaut. Bei den Transportbändern handelt es sich um kumulative Stetigförderer ([AF05]). Die individuellen Beschleunigungs- und Geschwindigkeitsverhältnisse der Transportbänder, der Umsetzkräne und der Werker wurden entsprechend den existierenden Zeitaufnahmen aus dem Bereich der Arbeitswirtschaft umgesetzt.

Über das gesamte Materialflusssystem hinweg sind Messpunkte verteilt. Vor und nach jeder der insgesamt 35 Montagestationen werden die Zeitstempel in einer zentralen Leitstandsdatenbank als Ein- und Auslaufdatum erfasst, die auf einem Oracle-DBMS basiert.

Weiterhin können aus dieser Datenbank die Stati der Maschinen und der Zeitstempel des letzten Statuswechsel ermittelt werden. Dies ist im Zusammenhang mit den Werkern nicht möglich, da personenbezogene Daten aus betriebsinternen Gründen nicht erfasst werden dürfen. Die Prozess-Sequenz und die Prozesszeiten der einzelnen Montagestationen lassen sich aus einem weiteren System ermitteln, welches auf einer DB2-Datenbank basiert und die Mensch- und Maschinenzeiten der einzelnen Arbeitsvorgänge im Materialflusssystem enthält.

Die Modellerstellung erfolgt mit Hilfe so genannter *Source Files*, die neben dem eigentlichen Modell (Anm.: das Modell repräsentiert das für jedes Materialflusssystem spezifische Prozess-System) Funktionalitäten zum Datenimport und -export, sowie zur Steuerung der Simulationsläufe enthalten. Darüber hinaus wurden die in Abschnitt 4.2.2 dargestellten Funktionalitäten realisiert.

Um einem Simulationsmodell in AutoMod die Informationen der einzelnen komplexen Kind-Elemente des XML-Schemas für Simulationsbasierte Frühwarnsysteme zur Verfügung zu stellen (vgl. Abschnitt 4.1.3), werden mit Hilfe einer XSLT-Transformation einfache ASCII-Files erzeugt. Abbildung 6.3 zeigt beispielhaft einen transformierten ASCII-File mit entsprechenden Ressourceninformationen des komplexen Kind-Elementes *Resources*.

Die Inhalte werden mit Hilfe von Einlesefunktionen verarbeitet und in Form von Listen, Variablen und Feldern verwaltet. Anschließend stehen sie der Steuerung und den einzelnen Funktionskomponenten zur Verfügung.

```

TRANSFORMED RESOURCES VALUES
*****
ResourceID      Description                                     State
*****
R_Station0      Setkorb auf Montagewagen setzen.              Active
R_Station1      Presse Losrad 1. Gang                          Active
R_Station2      Presse Losrad 2. Gang                          Active
R_Station2_2    Ausweichstation R_Station2                    Active
R_Station3      Presse Losrad 3. Gang                          Idle

```

Abb. 6.3: Transformiertes ASCII-File mit Ressourceninformationen

Im nachfolgenden Abschnitt wird auf die Ausnahmedetektion und Handlungsalternativengenerierung in der FSG-Montage eingegangen.

6.2.1.3 Ausnahmedetektion und Handlungsalternativengenerierung

Neben den üblichen Zielsetzungen einer Produktionssteuerung wie geringe Bestände und eine hohe Auslastung stehen bei der FSG-Montage folgende Hauptziele im Vordergrund:

1. Vermeidung von Verspätungen

Die FSG-Montage versorgt nachgelagerte Montagebereiche, die auf eine pünktliche Lieferung der Getriebe angewiesen sind. Die Reihenfolge und die Losgröße der abwechselnd zu montierenden Diesel- und Benzingetriebe wird vom Leitstand generiert.

2. Adäquate Auslastung der Werker

Eine zu hohe bzw. zu niedrige Auslastung ist zu verhindern. Dabei kann eine relativ kurzfristige Erhöhung und Verringerung der Werkeranzahl für die Folgeschicht oder eine Umverteilung mit Hilfe der vorgeschalteten angrenzenden Fertigungsbereiche erfolgen.

3. Ausfall einer Montagestation

Im ersten Montagekreislauf befinden sich insgesamt 14 Montagestationen, die zur Verpressung der Räder auf Wellen dienen. Fällt eine dieser Stationen aufgrund technischer Probleme aus, würde die gesamte Montagelinie zum Stillstand kommen. Aus diesem Grund existiert eine mobile Presse, die als *Ausweichstrategie* verwendet werden kann. Sie verhindert den vollständigen Produktionsstillstand und verschafft der Instandhaltung genügend Zeit, um das technische Problem zu beheben.

Anhand dieser Zielsetzungen lassen sich Ausnahmedefinitionen ableiten.

Definition von Ausnahmefällen

Die Vermeidung von Verspätungen ergibt die erste Ausnahmedefinition *i*. Hierfür ist den einzelnen Produkten ein Attribut Δt_V hinzuzufügen, welches sich aus der Differenz zwischen dem geplanten Montageende t_P und dem tatsächlichen Absetzzeitpunkt t_C aus dem letzten Montagekreislauf ergibt, mit $\Delta t_V = t_P - t_C$. Ein Ausnahmefall ist dann gegeben, wenn bei der Prognose für ein Produkt $\Delta t_V < 0$ detektiert wird.

Die adäquate Auslastung der Werker führt zu einer weiteren Ausnahmedefinition *j*. Dabei wird

sowohl eine obere und untere Schranke für die durchschnittliche Auslastung der Werker festgelegt. Bei der Ausnahmedetektion werden die Auslastungen aller Werker zur Berechnung der durchschnittlichen Auslastung herangezogen.

Die Verwendung einer mobilen Montagepresse als Ausweichstrategie bei einem Stationsausfall stellt den Ausnahmefall k dar, bei dem es sich um einen Sonderfall handelt (Monitoringfunktionalität, vgl. Abschnitt 3.2.2).

Handlungsalternativen in Abhängigkeit der Ausnahmefälle

Die Vermeidung von Verspätungen kann über eine Belegungsplanung oder die Veränderung der Werkeranzahl im Materialflusssystem erfolgen.

Ein Belegungsplanungsalgorithmus zieht dabei das verspätete Getriebe so weit in der Reihenfolge vor, dass die Verspätung vermieden wird (Einflussnahme über einen zeitlichen Steuerparameter, vgl. Abschnitt 5.1.2). Dabei ist das Auftreten neuer Verspätungen zu verhindern. In der Mappingmatrix werden für die Ausnahme i die Restriktionen *Auftragspriorisierung* und als *Optimalitätskriterium* die *Verspätung* markiert (vgl. Tab. 5.1).

Zur Vermeidung des Ausnahmefalls i kann der quantitative Steuerparameter *Werkeranzahl* erhöht werden (diese ist in dem Intervall [5; 8] frei wählbar), wodurch die Kapazität bzw. Produktionsgeschwindigkeit der FSG-Montage steigt und eine Verringerung der Verspätung eintritt. Darüber hinaus können Schichtpausen verkürzt werden, wobei eine Verringerung in 10 min-Schritten auf die gesamte Pausenzeit einer Schicht verteilt wird. Weiterhin ist das Schichtmodell zwischen 2 und 3 Schichten frei wählbar.

Auch der Ausnahmefall j ist ebenfalls durch die Veränderung der Werkeranzahl beeinflussbar. Bei einer Überschreitung der Auslastung ist die Verringerung der Werkeranzahl geeignet, während bei einer Unterschreitung eine Erhöhung stattfinden muss. Da in der Koeffizientenmatrix nur eine Veränderungsrichtung (positiv oder negativ) angegeben werden kann, wird der Ausnahmefall j in zwei Ausnahmefälle aufgetrennt. Dabei repräsentiert der Ausnahmefall j_U die Überschreitung, während $j_{\bar{U}}$ die Unterschreitung darstellt.

Dem letzten Ausnahmefall k entspricht der Sonderfall einer mobilen Montagepresse.

Tabelle 6.1 stellt die Korrelationskoeffizienten der einzelnen Ausnahmen und Handlungsalternativen in der Koeffizientenmatrix dar. Dabei werden die Koeffizienten $c_{1,i}$, $c_{2,i}$, $c_{3,i}$ und c_{1,j_U} auf 0.5, $c_{1,j_{\bar{U}}}$ auf -0.5, sowie $c_{4,k}$ auf 1.0 gesetzt.

Quantitative Einflussparameter / Sonderfälle	Ausnahmefälle			
	Ausnahme i : Aufträge zu spät	Ausnahme j_U : Ø Werker- auslastung überschritten	Ausnahme $j_{\bar{U}}$: Ø Werker- auslastung unterschritten	Ausnahme k : Stationsausfall
1 : Werkeranzahl(c): c=5;6;7;8	0,50	0,50	-0,50	0,00
2 : Pausenzeiten(c): c=10;20;30	-0,50	0,00	0,00	0,00
3 : Schichtmodell(c): c=2,3	0,50	0,00	0,00	0,00
4 : Sonderfall Presse	0,00	0,00	0,00	1,00

Tab. 6.1: Korrelationskoeffizienten für die FSG-Montage

Bewertung des pilotiven Einsatzes in der FSG-Montage

Die FSG-Montage ist ein Materialflusssystem mit überschaubarer Komplexität, da nur zwei Getriebevarianten montiert werden. Weiterhin sind Messpunkte über das gesamte Materialflusssystem verteilt und es erfolgt eine zentrale und automatisierte Erfassung der Zeitstempel.

Die tatsächlichen Taktzeiten von 2,6 bis 3,4 Minuten je Getriebe stellen hohe Anforderungen an die Performance des SEWS. Die Auswertungen der Zwischenankunftszeiten der Zustandsänderungen ergaben bei einem Konfidenzintervall von 90 % eine feste Zeitspanne von 30 Sekunden zwischen zwei Prognosen (vgl. Abschnitt 4.3.2.2).

Weiterhin problematisch ist die gegenwärtig nicht automatisierbare Rückführung der Ergebnisse einer Belegungsplanung eines SEWS in den realen Leitstand. Wird eine Handlungsalternative in Realität umgesetzt, erfolgt keine Aktualisierung des Leitstandes, wodurch künftige Prognosen auf unstimmigen Daten durchgeführt werden. Dies ist nur durch eine manuelle Rückkopplung möglich.

6.2.1.4 Ergebnisse des SEWS für die FSG-Montage

Die Ergebnisse der Prognosen eines SEWS weichen von den Daten der Realität ab. Die Berechnung der Messpunktabweichung ist durch den Vergleich von auf Basis historischer Daten getroffene Prognosen mit den tatsächlich aufgetretenen Daten möglich (vgl. Abschnitt 4.3.3). Eine Auswertung der mittleren absoluten Abweichung (MAD) der Zeitstempel der Messpunkte wird in Abbildung 6.4 dargestellt. Dabei wurde ein repräsentativer Zeitraum von zwei Schichten eines Produktionstages ausgewählt, in deren Verlauf keine Besonderheiten wie etwa Stationsausfälle oder kurzfristige Veränderungen der Belegungsplanung aufgetreten sind. Die MAD ergibt sich unter der Verwendung des Prognosewertes des Zeitstempels $t_{i,j}^P$ eines Produktes i am Messpunkt j und des tatsächlichen Wertes in den Systemen $t_{i,j}^V$ zu

$$MAD = \frac{1}{n \cdot m} \sum_{i=1}^n \sum_{j=1}^m |t_{i,j}^P - t_{i,j}^V|.$$

Direkt nach der Initialisierung existiert keine Abweichung, die aber mit fortschreitender Zeit kumulativ zunimmt. Je früher im Materialflusssystem eine Abweichung auftritt, desto negativer wirkt sich diese aus.

Zur Unterscheidung und genaueren Analyse der verschiedenen Einflüsse auf die Abweichung erfolgt eine Aufteilung in drei Abschnitte, die den einzelnen Montagekreisläufen der FSG-Montage entsprechen (vgl. Abschnitt 6.2.1.1). Der letzte Montagekreislauf besitzt aufgrund seiner weitestgehenden Automatisierung einen geringen Einfluss auf die Messpunktabweichung. Dagegen wirken sich der erste und zweite Montagekreislauf wesentlich stärker aus, woran der erste Montagekreislauf aufgrund der erhöhten Anzahl an Messpunkten einen größeren Anteil hat.

Bei einer genaueren Analyse ergibt sich ein interessanter Zusammenhang bezüglich des Beginns und Endes einer Schicht, sowie Pausenzeiten. Hier erhöht sich die Messpunktabweichung, da die exakte Einhaltung des Pausenbeginns und -endes bzw. Schichtwechsels in Realität nicht der Fall ist.

Innerhalb der ersten Schicht liegt die Abweichung bei ca. 3,5 Minuten. Wird der Prognosezeitraum auf zwei Schichten ausgeweitet, steigt die Abweichung auf bedenkliche 10 Minuten an. Überraschenderweise ergab sich, dass die Belegungsplanung einen geringen Einfluss auf

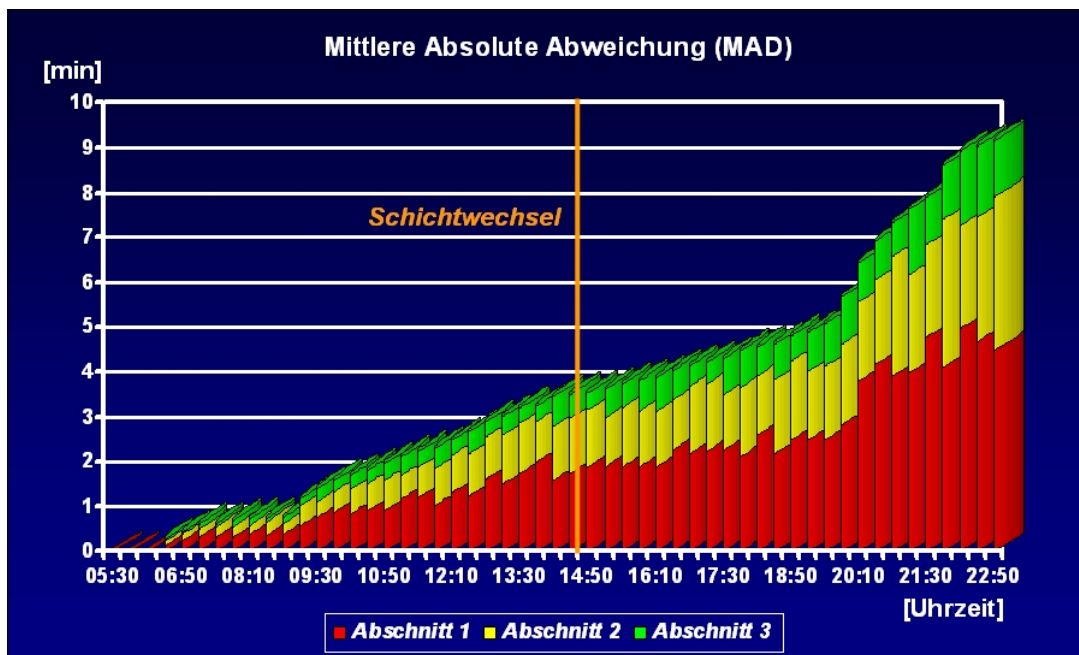


Abb. 6.4: Auswertung der Mittleren Absoluten Abweichung für die FSG-Montage

die Ausnahmefälle hat. Da die durch den Leitstand erzeugte Belegungsplanung die Termine des Montageendes in der Reihenfolge der Abliefertermine generiert, verursacht das Vorziehen eines Getriebes die Verspätung anderer Getriebe.

Die quantitativen Steuerparameter Werkeranzahl, Pausenzeiten und das Schichtmodell eignen sich besser zur Vermeidung von Verspätungen. Dabei stellt die Werkeranzahl und die Veränderung von Pausenzeiten die effizienteste dieser Möglichkeiten dar. Eine Veränderung des Schichtmodells führt zur Verschlechterung der Werkerauslastung und erzeugt somit vermehrt den Ausnahmefall j_U (vgl. Abschnitt 6.2.1.4).

Durch den Einsatz eines SEWS in der FSG-Montage kann der Werkereinsatz verbessert werden, da eine zu hohe oder zu geringe Auslastung verhindert wird. Weiterhin sind Verspätungen durch eine Erhöhung der Werkeranzahl bzw. Verringerung der Pausenzeiten schon frühzeitig vermeidbar. Hierfür liefert das SEWS geeignete Lösungsvorschläge, die auch praktisch umgesetzt werden können.

Abbildung 6.5 zeigt einen prognostizierten Verlauf für die durchschnittliche Werkerauslastung in der FSG-Montage. Dabei ist aus den Prognosedaten erkennbar, dass die Werkerauslastung zu Beginn der zweiten Schicht den oberen Grenzwert überschreitet. Das SEWS schlägt als Handlungsalternative den Einsatz eines zusätzlichen Werkers vor, wodurch der Ausnahmefall verhindert wird.

Im Folgenden wird der pilotive Einsatz von SEWS in einem weiteren Materialflusssystem vorgestellt.

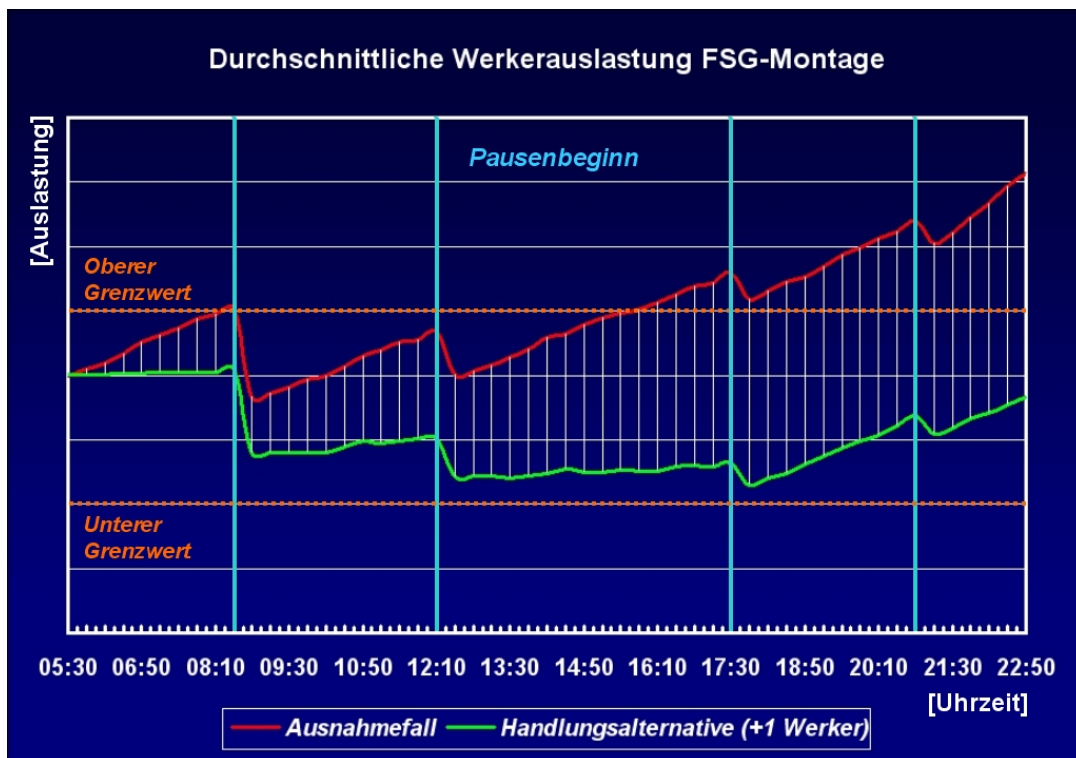


Abb. 6.5: Auswertung der durchschnittlichen WerkerAuslastung in der FSG-Montage

6.2.2 Der Härteofen A28

Der Härteprozess ist ein wesentlicher Bestandteil des Produktionsprozesses in der Getriebeherstellung. Hierfür kommen Härteöfen zum Einsatz, die Wellen und Räder in einem Temperaturbereich von 880 bis 980 °C härten. Es handelt sich dabei um einen für die Qualität und Lebensdauer eines Getriebes sehr bedeutsamen Prozess, der als *Einsatzhärten* bezeichnet wird. Im Allgemeinen wird unter dem Einsatzhärten das Aufkohlen, Härten und Anlassen eines Werkstückes aus Stahl verstanden ([Gro94]).

Als Repräsentant eines dieser Materialflusssysteme wurde der Härteofen A28 des Getriebeherwerkes Rastatt der DaimlerChrysler AG ausgewählt. Es handelt sich hierbei um einen Ringhärteofen, der in so genannten *Rosten* befindliche Wellen und Räder aufnimmt und härtet.

6.2.2.1 Beschreibung des Materialflusssystemes

Der Ringhärteofen A28 wird über Zuführ- und Abfuhrbänder mit zu härtendem Material versorgt. Ein Vorwärmofen verhindert zu große Spannungen beim Einbringen des Materials in den eigentlichen Ringhärteofen. Nach dem Härteofen werden die Roste in ein Ölbad zum Abschrecken verfahren. Eine Waschmaschine reinigt anschließend die Wellen und Räder.

Der eigentliche Ringhärteofen kann insgesamt 36 Roste aufnehmen, die mit einer bestimmten Taktzeit durch den Ofen gefahren werden. Die Taktzeit ist dabei sowohl von dem sich gerade

im Ofen befindenden als auch von dem als nächstes zu härtenden Materialtyp abhängig. Zu diesem Zweck wird der Begriff *Ofenlos* verwendet. Ein Ofenlos besteht aus einer bestimmten Menge an Rosten eines Materialtyps. Insgesamt sind drei Materialtypen zu unterscheiden, die im Folgenden als *Vorgelegewellen (VW)*, *Hauptwellen (HW)* und *Antriebswellen (AW)* bezeichnet werden. Die Aufenthaltszeit des jeweiligen Materialtyps ergibt sich dabei aus dessen Taktzeit und der Anzahl an Stellplätzen im Härteofen.

Um in der betrieblichen Praxis die Steuerung zu vereinfachen, kommt zur Ofensteuerung eine feste Ofenbelegung zum Einsatz. Diese ist anhand der zu produzierenden Stückzahlen in den Montagen festgelegt, kann allerdings bei Bedarf flexibel umgestellt und anschließend wieder in die feste Reihenfolge überführt werden.

Während des Härtevorgangs können sich die Werkstücke verziehen. Mit Hilfe so genannter Richtbänke wird dieser Mangel durch Biegen und Pressen behoben, wobei der Vorgang selbst als *Richten* bezeichnet wird und mit einer Prozesszeit und Rüstzeiten verbunden ist. Nach dem Richten werden die gehärteten Materialien in einen Puffer verbracht, aus dem sich die nachgelagerten Fertigungsbereiche bedienen können. Abbildung 6.6 verdeutlicht den schematischen Aufbau des Materialflusssystemes Härteofen A28.

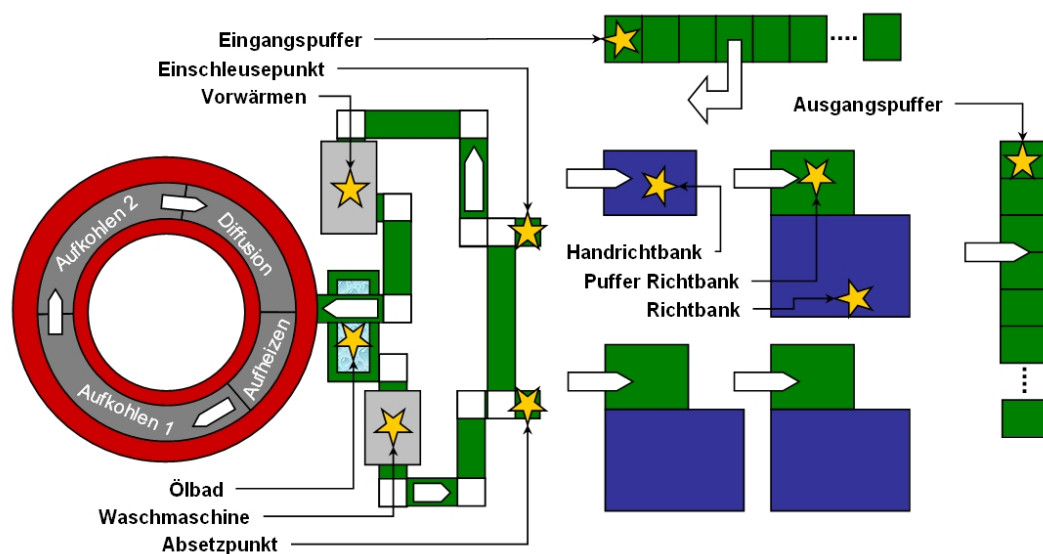


Abb. 6.6: Schematisches Layout des Härteofens A28

Der Härteofen A28 wird in *kontinuierlicher Schichtarbeit (Konti-Schicht)* mit Pausenüberbrückung betrieben. Dies bedeutet, dass der Ofen 24 Stunden am Tag und das sieben Tage die Woche produktiv ist und Material härten kann.

6.2.2.2 Simulationsmodell und Datenversorgung

Für den Härteofen A28 wurde ein Simulationsmodell mit dem Simulator SLX entwickelt. In der SLX-Entwicklungsumgebung erfolgt eine Darstellung des Prozess-Systems, der Transport-

bänder, des Härteofens, der Richtmaschinen und der Puffer mit Hilfe vorgegebene Elemente der Sprache SLX wie bspw. *Queues*, *User Chains*, *Facilities* und *Storages* ([SH98]). Die Geschwindigkeits- und Beschleunigungsverhältnisse der realen Objekte sind entsprechend berücksichtigt worden. Die Visualisierung des Modells erfolgte mit Hilfe des Animationswerkzeugs *Proof Animation*, der eine zweidimensionale Darstellung der Simulationsläufe ermöglicht. Wie bei der FSG-Montage wurde eine Layoutdatei im DXF-Format herangezogen, um die reale Dimensionierung des Materialflusssystem zu berücksichtigen.

Auch beim Härteofen A28 sind eine Vielzahl von Messpunkten vorhanden, die automatisiert Daten an den so genannten *Ofenleitstand* zurückmelden. Die Ein- und Ausgangsbereiche der Richtbänke werden entsprechend erfasst. Eine manuelle Rückmeldung erfolgt bei der Verbringung von gerichtetem Material in den Ausgangspuffer und zu härtendes Material in den Eingangspuffer des Härtereibereiches. Dadurch ist die gegenwärtige Position eines Rostes im Materialflusssystem jederzeit feststellbar. Dies gilt auch für die Stati der einzelnen Roste, der Heber, Transportbänder und Richtmaschinen. Wie bei der FSG-Montage ist eine explizite Erfassung von Werkerstati nicht möglich. Die Verwaltung der Prozess-Sequenz und der Prozesszeiten ist identisch (vgl. Abschnitt 6.2.1.2).

Die Steuerung des Simulationsmodells und die Funktionskomponenten werden in Modulen hinterlegt und in der SLX-Sprache programmiert. Abgespeichert wird das Modell in einer *slx*-Datei und muss vor der Ausführung kompiliert werden. Die Animation wiederum wird mit Hilfe so genannter *Animation Trace Files (ATF)* beschrieben, die vom Proof Animator ausführbar sind.

Wie für das Simulationsmodell der FSG-Montage liefert das Framework des SEWS alle Informationen der komplexen Kind-Elemente des XML-Schemas für Simulationsbasierte Frühwarnsysteme. Diese werden per XSLT-Transformation in ein entsprechendes Format gebracht und in ASCII-Files abgespeichert (vgl. Abb. 6.3). Auch in SLX existieren Einlesefunktionen, mit deren Hilfe die Inhalte der transformierten ASCII-Files eingelesen und in die Simulationsmodell-interne Verwaltungsstruktur übertragen werden.

Im nächsten Abschnitt erfolgt die Erläuterung der Ausnahmedefinition und Handlungsalternativengenerierung.

6.2.2.3 Ausnahmedetektion und Handlungsalternativengenerierung

Für das Materialflusssystem A28 sind neben einem effizienten Materialfluss mit hoher Auslastung, geringen Rüstungen und einer hohen Verfügbarkeit vor allem folgende Zielsetzungen von Bedeutung:

1. Sicherung der Versorgung nachfolgender Bereiche

Nachfolgende Fertigungsbereiche sind von der Versorgung mit Teilen aus der Härterei abhängig. Dabei treten täglich Produktionsschwankungen auf, die durch einen ausreichend großen Mindestbestand im Ausgangspuffer der Härterei auszugleichen sind.

2. Keine Überschreitung des maximalen Bestandes in der Härterei

Da die Richtbänke zur Bearbeitung verschiedener Materialtypen zu rüsten sind, müssen ausreichend Pufferplätze zur Verfügung gestellt werden. Zwischen dem Härteofen und den Richtbänken ist nur eine beschränkte Puffermöglichkeit vorhanden.

3. Ausfall einer Richtbank

Der Ausfall einer Richtbank führt zwangsläufig zur Erhöhung des Bestandes in der Härterei und zu einer unsicheren Versorgung nachfolgender Bereiche. Dies kann frühzeitig durch eine Belegungsplanung oder eine *Ausweichstrategie* in Form einer Handrichtbank ausgeglichen werden.

Aus diesen Zielsetzungen ergeben sich wiederum entsprechende Ausnahmefalldefinitionen.

Definition von Ausnahmefällen

Die erste Ausnahmefalldefinition ergibt sich aufgrund der Forderung nach minimalen Pufferbeständen der einzelnen Teilenummern im Ausgangspuffer. Bei insgesamt n unterschiedlichen Teilen ergeben sich somit n Ausnahmefalldefinitionen mit der Verknüpfung minimaler Bestände p_i für jedes Teil i , mit $i = 1, \dots, n$.

Eine weitere Ausnahmefalldefinition j ergibt sich bezüglich eines Maximalbestandes an Rosten zwischen dem Ausgang des Härteofens und den Richtbänken.

Der Ausnahmefall k repräsentiert die Ausweichstrategie bei Ausfall einer Richtbank. Sie benötigt eine Belegungsplanung mit veränderter Ressourcenkapazität durch zuschalten der Handrichtbank. Die Handrichtbank kann aber auch ohne Durchführung einer Belegungsplanung verwendet werden. Im Falle des Ausnahmefalls k wird angenommen, dass die entsprechende Richtbank für den gesamten Prognosezeitraum ausfällt.

Handlungsalternativen in Abhängigkeit der Ausnahmefälle

Zur Handlungsalternativengenerierung sind nun die entsprechenden Informationen für die zeitlichen, qualitativen und quantitativen Steuerparameter in der Mapping- und Koeffizientenmatrix zu hinterlegen. Bezüglich der Kriterien zur Belegungsplanung werden die entsprechenden Eintragungen in der Mappingmatrix in Abbildung 6.2 dargestellt.

Mapping	Ausnahmefälle					
	Ausnahme 1: Bestand Teil 1 zu niedrig	Ausnahme 2: Bestand Teil 2 zu niedrig	...	Ausnahme n: Bestand Teil n zu niedrig	Ausnahme j: Max-Bestand überschritten	Ausnahme k: Ausfall einer Richtbank
Ressourcenverfügbarkeit						X
Kapazitätsbeschränkungen					X	
Auftragspriorisierung	X	X	X	X		
Optimalitätskriterium						
Durchlaufzeit						
Wartezeit						
Leerzeit						
Rüstzeit					X	X
Terminabweichung						
Verspätung	X	X	X	X		

Tab. 6.2: Mappingmatrix für den Härteofen A28

Die Belegungsplanung zur Vermeidung der Ausnahmefälle 1 bis n erfolgt durch einen Algorithmus, der die einzelnen Materialien innerhalb der festen Reihenfolge der Ofenbelegung verändert. Hierzu wird eine einfache Verschiebung von Rosten vorgenommen. Der Übergabepara-

meter entspricht dabei der Verspätung des Teiletyps, der sich aus der Differenz des Zeitpunktes der Unterschreitung des Minimalbestandes und der prognostizierten Ankunft der Teile ergibt. Dieser Algorithmus gestaltet sich relativ einfach, da eine feste Reihenfolge verfolgt werden kann. Hierbei werden die Positionierung der Teilenummern innerhalb der festen Reihenfolge und Ofenlose verändert, und es wird überprüft, ob dadurch die Verspätung ausgeglichen werden kann. Liefert dies keine Lösung, versucht der Algorithmus, die Losgrößen zu verändern und Roste einem früheren Los *anzuhängen*, ohne weitere Verspätungen zu verursachen. Der Belegungsplanungsalgorithmus des Ausnahmefalls j versucht, den Bestand zwischen dem Ofenausgang und den Richtbänken durch Verringerung der Anzahl Rüstungen so gering wie möglich zu halten.

Für den Ausnahmefall k wird der gleiche Algorithmus wie für den Ausnahmefall j eingesetzt, wobei die Zuschaltung der Handrichtmaschine berücksichtigt wird.

Das vorrangige Ziel dabei ist eine Rüstoptimierung, indem wiederum der Aufbau und die Größe des Ofenloses so verändert wird, dass möglichst große *Packungen* gleicher Teiletypen entstehen.

Der einzige quantitative Steuerparameter bezüglich der obigen Ausnahmefälle stellt der Einsatz der Handrichtbank dar, der mit zusätzlichem Werkereinsatz verbunden ist. Dieser Sonderfall kann ebenfalls zur Vermeidung der Ausnahmefälle j und k eingesetzt werden, ohne eine Belegungsplanung durchzuführen. Hierzu werden die Korrelationskoeffizienten $c_{1,j}$ und $c_{1,k}$ in der Koeffizientenmatrix auf 0.5 gesetzt.

Bewertung des pilotiven Einsatzes

Das Materialflusssystem Härteofen A28 ist aufgrund seiner Überschaubarkeit und dem festen Ofenprogramm besonders für einen pilotiven Einsatz geeignet. Dabei sind große Potenziale durch die Komplexität der Umstellung und Veränderung des Ofenloses vorhanden, das bei einem prognostizierten Ausnahmefall durch das SEWS angepasst wird. Diese Funktionalität wird bisher nicht durch den Ofenleitstand zur Verfügung gestellt. Im Vergleich mit der FSG-Montage tritt die Belegungsplanung als qualitativer und zeitlicher Steuerparameter stärker in den Vordergrund (vgl. Abschnitt 6.2.1.3).

Die vergleichsweise langen Taktzeiten (durchschnittlich 15 Minuten), Rüst- (mindestens 30 Minuten) und Richtzeiten (mindestens 10 Minuten) ermöglichen a priori die Festlegung einer Zeitspanne zwischen zwei Prognosen, für die $\Delta t_{Det} = 5$ Minuten gilt. Dadurch ist die Performance des Belegungsplanungsalgorithmus keine kritische Größe.

Wie bei der FSG-Montage (vgl. Abschnitt 6.2.1.3) erfolgt keine automatisierte Rückkopplung zum realen Leitstandssystem, wodurch die Genauigkeit künftiger Prognosen verschlechtert wird.

6.2.2.4 Ergebnisse des SEWS für den Härteofen A28

Wie in Abschnitt 4.3.3 bereits erläutert wird, weichen die Prognosen von der Realität mehr oder weniger stark aus den verschiedensten Gründen ab. Ein geeignetes Maß stellt die Messpunktabweichung dar, die auch für dieses reale Materialflusssystem herangezogen wird.

Abbildung 6.7 zeigt eine Auswertung der mittleren absoluten Abweichung (MAD) der Zeitstempel an den Messpunkten. Verwendet wurde hierfür ein repräsentativer Zeitraum, der keine Besonderheiten wie Maschinenausfälle oder Veränderungen der Belegungsplanung aufweist. Dabei ist erkennbar, dass zum Prognosebeginn direkt nach der Initialisierung keine Abweichung besteht, die allerdings im Verlauf des Prognosezeitraums zunimmt. Die MAD kumuliert mit der zunehmenden Anzahl an Abweichungen, die sich bei längeren Prognosezeiträumen verstärkt häufen.

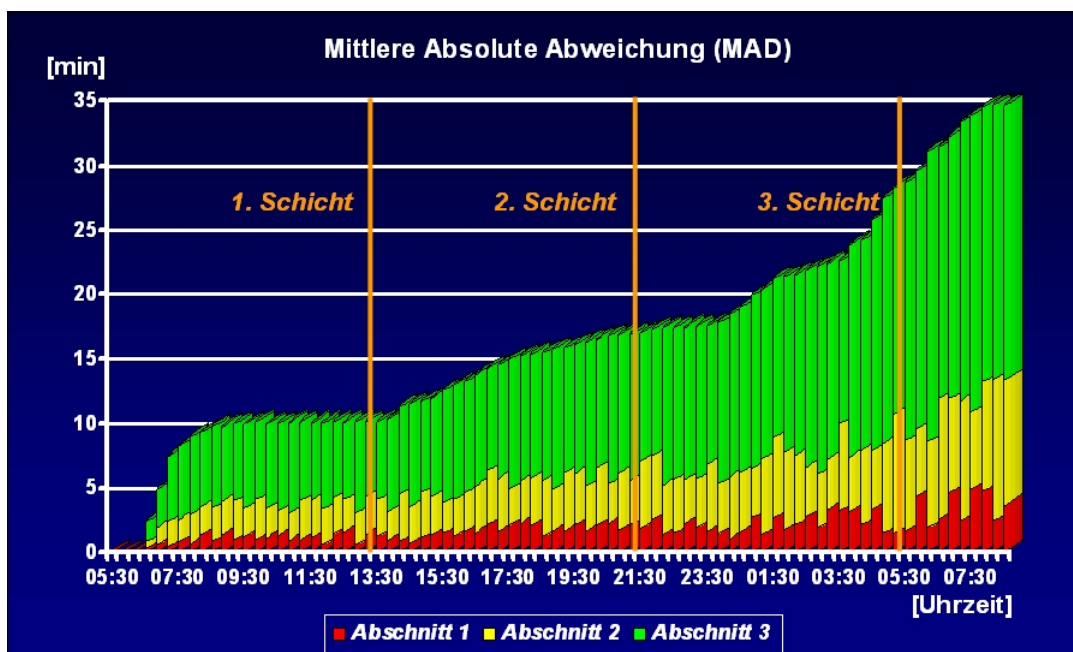


Abb. 6.7: Auswertung der Mittleren Absoluten Abweichung des Härteofens A28

Dies hat mehrere Gründe, die sich auf die manuellen Tätigkeiten im Materialflusssystem zurückführen lassen. Eine Unterteilung der Messpunktabweichung in drei Abschnitte entsprechend dem Aufbau des Materialflusssystems soll die Zusammenhänge verdeutlichen:

- **Abschnitt 1**
Dieser Abschnitt erstreckt sich von dem Eingangspuffer über den Ofen hinweg bis zum Absetzpunkt. In der Abbildung 6.7 ist erkennbar, dass dieser Abschnitt nur einen geringeren Einfluss auf die Messpunktabweichung hat.
- **Abschnitt 2**
Der zweite Abschnitt beginnt nach dem Absetzpunkt und endet beim Start der Bearbeitung auf den Richtbänken. Dabei werden die Roste manuell zu den Richtbänken gebracht, die jeweils von einer automatischen Beladung versorgt werden.
- **Abschnitt 3**
Der letzte Abschnitt reicht von der Beendigung des Richtens bis zur manuellen Ablieferung der gehärteten und gerichteten Teile im Ausgangspuffer. Dieser Vorgang ist mit einer manuellen Rückmeldung verbunden, die den größten Beitrag zur MAD liefert.

Die über einen Prognosezeitraum dargestellte Veränderung der MAD zeigt einen Maximalwert ca. 35 Minuten, der eine gesicherte Aussagefähigkeit unmöglich macht. Bei 16 Stunden beträgt dieser Wert etwa 17,5 Minuten, was annähernd dem durchschnittlichen Ofentakt entspricht und somit als ausreichend erachtet werden kann.

Das Auftreten der Unterschreitung der minimalen Bestände des Ausgangspuffer konnte mit Hilfe der Belegungsplanung in ca. 65 % der Fälle vermieden werden. Dabei wurde deutlich, dass die minimalen Bestände derjenigen Teiletypen zu gering gewählt waren, die geringere Stückzahlen aufwiesen. Nach einer Erhöhung der Minimalbestände der 50 % stückzahlschwächsten Teiletypen um 25 % ermöglichte die Belegungsplanung bereits eine Vermeidung der Bestandsunterschreitung in 75 % der Fälle.

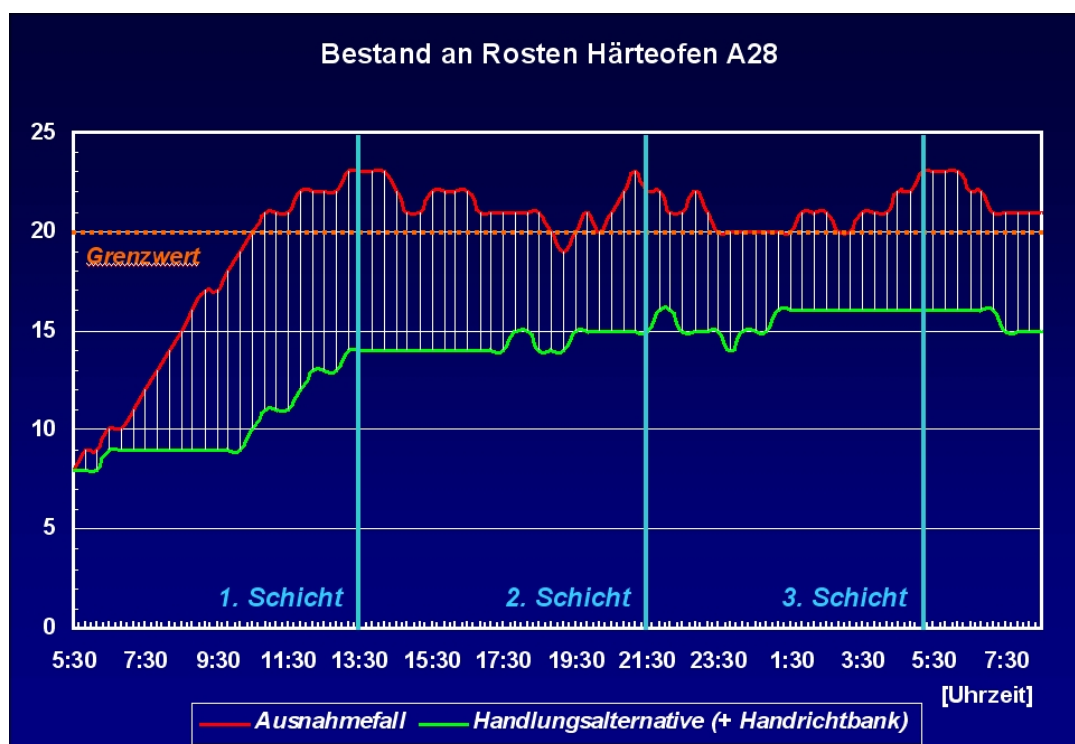


Abb. 6.8: Auswertung des Bestandes an Rosten im Materialflusssystem Härteofen A28

Der Ausfall einer Richtbank und die Überschreitung des Maximalbestandes in der Härterei sind seltene Ereignisse im realen Materialflusssystem. Diese können mit einem SEWS vollständig vermieden werden, da allein durch den zusätzlichen Einsatz der Handrichtbank die Kapazität beim Richten erhöht und somit ein zügiger Abbau der Bestandsüberschreitung möglich ist.

Im realen Materialflusssystem kann der Maximalbestand auf andere Puffer in angrenzende Bereiche verteilt werden. Dies zieht einen erhöhten Logistikaufwand nach sich und wurde im SEWS aufgrund fehlender Messpunkte und somit einer mangelhaften Nachvollziehbarkeit vernachlässigt.

Die Vorteile des pilotiven Einsatzes eines SEWS für diesen Härteofen liegen in dem Einsatz einer Belegungsplanung, die eine Verbesserung der Bestandssituation bewirkte. Dies war zuvor

nicht möglich und die Funktionsfähigkeit konnte eindeutig nachgewiesen werden. Weiterhin zeigte sich, dass die minimalen Bestände des Ausgangspuffers für stückzahlschwache Teiltypen zu gering gewählt waren.

Wie im Falle der FSG-Montage findet keine automatisierte Rückführung von Daten in den Ofenleitstand statt, wodurch die Ergebnisse nachfolgender Prognosen anzuzweifeln sind. Weiterhin führen manuelle Prozesse und Rückmeldepunkte relativ schnell zu einer starken Erhöhung der Messpunktabweichung, wodurch die freie Wählbarkeit der Länge des Prognosezeitraumes beschränkt ist.

In Abbildung 6.8 wird ein prognostizierter Bestandsverlauf von Rosten im Abschnitt 2 des Materialflusssystemes A28 dargestellt. Dabei weisen die Prognosedaten eine Überschreitung der kritischen Bestandsgrenze von 20 Rosten während der ersten Schicht und während dem weiteren Verlauf der Folgeschichten auf. Das SEWS schlägt als Handlungsalternative die *Handrichtbank* vor, deren Anwendung zur Vermeidung des Ausnahmefalls führt.

6.3 Zusammenfassung des Kapitels

In diesem Kapitel wird die praktische Umsetzbarkeit Simulationsbasierter Frühwarnsysteme diskutiert. Hierzu erfolgt eine Bewertung der technischen Umsetzung im betrieblichen Umfeld eines Industrieunternehmens. Dabei wird sowohl der Einsatz verteilter Systeme, Web Services und XML-Datenbanken als auch die praktischen Problemstellungen bei der Datenbeschaffung und -versorgung, sowie die verwendeten Simulatoren betrachtet. Anschließend erfolgt die Vorstellung von Fallbeispielen, die in Form der FSG-Montage und dem Härteofen A28 eine Bewertung des Einsatzes Simulationsbasierter Frühwarnsysteme ermöglichen. Diese Beispiele werden explizit beschrieben, wobei jeweils die Datenversorgung, die Simulationsmodelle, die Ausnahmedetektion und Handlungsalternativengenerierung erörtert, sowie die konkreten Ergebnisse diskutiert werden.

Dieses Kapitel befasst sich mit der prototypischen Umsetzung von SEWS im betrieblichen Alltag eines Industrieunternehmens im Allgemeinen bzw. eines Automobilkonzerns im Besonderen. Die in den vorangehenden Kapiteln vorgestellten theoretischen Ansätze werden hinsichtlich ihrer Praxistauglichkeit bewertet. Die Datenbeschaffung und -versorgung wird durch die überwiegende Anzahl der auf modernen Datenbanktechnologien basierenden betrieblichen IT-Systeme erleichtert. Die verwendeten Simulatoren unterstützen die technische Umsetzung eines SEWS entsprechend der in den Abschnitten 3.3 und 3.4 vorgestellten Architektur.

Die Materialflusssysteme FSG-Montage und Härteofen A28 repräsentieren wichtige Teilprozesse der Getriebeproduktion bei der DaimlerChrysler AG. Dadurch lassen sich sowohl die Anwendbarkeit Simulationsbasierter Frühwarnsysteme im betrieblichen Umfeld nachweisen als auch die noch vorhandenen Mängel aufzeigen.

Kapitel 7

Zusammenfassung und Ausblick

*Die Natur hat ebenso das Ende eines jeden Dinges zum Ziel
wie seinen Anfang oder seine Fortsetzung, gleichsam wie der,
der einen Ball aufwirft.*
— Marc Aurel

7.1 Zusammenfassung der Arbeit

In einem ersten Kapitel wird auf die Computersimulation eingegangen. Dabei sind zunächst die geschichtliche Entwicklung, grundsätzliche Rahmenbedingungen für deren Einsatz, sowie Definitionen wichtiger Begrifflichkeiten interessante Themenschwerpunkte. Weiterhin erfolgt sowohl eine Klassifizierung der Simulation und eine Vorstellung grundsätzlicher Methoden als auch die Darstellung von Anwendungsbereichen, sowie deren Nutzen und Grenzen. Anschließend wird die Simulation im Kontext der Automobilindustrie betrachtet. Deren Haupteinsatzgebiete liegen sowohl in der Planung von Produktionsanlagen, Logistikbereichen und -konzepten als auch in der operativen Produktionsplanung und -steuerung. In diesem Zuge sind besonders die informationstechnische Integration der Simulation in einem industriellen Umfeld, der Begriff der Digitalen Fabrik und der Einsatz der Simulation speziell in den Bereichen Produktion und Logistik von Interesse. Auf das Themengebiet der Simulationswerkzeuge wird ebenso eingegangen wie auf die Vorgehensweise bei der Modellerstellung. Zum Abschluss werden Entwicklungstendenzen der Simulation in Produktion und Logistik diskutiert.

Ein weiteres Kapitel beschäftigt sich mit der Thematik der Simulationsbasierten Frühwarnsysteme. Hierzu werden allgemeine Frühwarnsysteme anhand ihrer grundlegenden Funktionalitäten erläutert und es wird auf strategische Frühwarnsysteme in Unternehmen eingegangen. Anschließend befasst sich das Kapitel speziell mit Simulationsbasierten Frühwarnsystemen in Produktion und Logistik. Es existiert noch keine geeignete und umfassende Definition in der Literatur. Aus diesem Grund werden eine Definition hergeleitet und die an SEWS zu stellenden Anforderungen entwickelt, wonach eine Abgrenzung und Einordnung im Vergleich zu anderen simulationsbasierten Ansätzen erfolgt. Anschließend beschäftigt sich das Kapitel mit einer allgemeinen Architektur Simulationsbasierter Frühwarnsysteme, wozu deren einzelne Komponenten vorgestellt werden. Diese Komponenten übernehmen dabei vielfältige Funktionalitäten, die sich zum einen aus den gestellten Anforderungen und zum anderen aus der notwendigen Integration der Hauptkomponenten zu einer Gesamtarchitektur ergeben. Es wird auf verteilte

Systeme und Kommunikationsprinzipien eingegangen, die bei der Entwicklung einer verteilten Architektur anwendbar sind. Web Services respektive eine serviceorientierte Architektur sind hierbei geeignete Technologien. Abschließend wird auf den Workflow Simulationsbasierter Frühwarnsysteme und deren Steuerungsmöglichkeiten eingegangen.

Das dritte Kapitel diskutiert die Prognose künftiger Zustände des realen Materialflusssystems. Hierzu wird zunächst die Datenbeschaffung und -verwaltung betrachtet, wobei eine Unterscheidung von simulations- und SEWS-relevanten Daten erfolgt. Anschließend beschäftigt sich das Kapitel mit Kriterien zur Systemauswahl, die im Zusammenhang mit der Datenbeschaffung bei der Auswahl geeigneter Informationssysteme und Datenquellen Anwendung finden. Auf die SEWS-interne Datenverwaltung mit XML und einer XML-Datenbank wird ebenso eingegangen, wie auf die eigentliche Datengewinnung aus originären Datenquellen. Die Verwendung der Simulation als eingebettete Applikation führt zu speziellen Anforderungen an SEWS. Bei der Prognose künftiger Zustände des realen Systems steht die Problematik der Initialisierung von Simulationsmodellen und der technischen Umsetzung der Detektion von Ausnahmefällen im Vordergrund. Weiterhin werden sowohl Möglichkeiten zur Modellaktualisierung als auch das Problem der Prognosegenauigkeit diskutiert.

Das folgende Kapitel befasst sich mit der Generierung von Handlungsalternativen im Ausnahmefall, wozu die hierfür generelle Vorgehensweise dargestellt wird. Die Schwerpunkte hierbei sind der Ablauf der Handlungsalternativengenerierung, die generellen Steuerparameter eines realen Materialflusssystems und die Begründung der Notwendigkeit eines Lösungsverfahrens. Weiterhin wird die Einordnung und Abgrenzung der Problemstellung der Handlungsalternativengenerierung bei Auftreten eines Ausnahmefalls diskutiert. Hierzu greift das Kapitel die Technologien Fuzzy-Logik, Künstliche Neuronale Netze, Regelbasierte Systeme und Expertensysteme, sowie die Thematik der Simulation und Optimierung auf. Anschließend erfolgt die Darstellung eines Verfahrens zur Ermittlung und Bewertung von Handlungsalternativen für Simulationsbasierte Frühwarnsysteme. Dabei wird eine mehrstufige Vorgehensweise begründet, die Darstellung von Handlungsalternativen in der SEWS-internen Datenverwaltung erläutert und sowohl auf die zeitliche und qualitative Belegungsplanung als auch auf quantitative Steuerparameter und Sonderfälle eingegangen. Weiterhin erfolgt die Vorstellung eines allgemeinen Algorithmus zur Handlungsalternativengenerierung und die Adaption der Korrelationskoeffizienten an ein Materialflusssystem. Zum Abschluss werden Abbruchkriterien eingeführt, die eine Beendigung des Algorithmus garantieren.

Ein letztes Kapitel diskutiert die prototypische Anwendbarkeit Simulationsbasierter Frühwarnsysteme. Hierzu erfolgt eine Bewertung der technischen Umsetzung im betrieblichen Umfeld eines Industrieunternehmens. Dabei wird sowohl der Einsatz verteilter Systeme, Web Services und XML-Datenbanken als auch die praktischen Problemstellungen bei der Datenbeschaffung und -versorgung, sowie die verwendeten Simulatoren betrachtet. Es erfolgt die Erläuterung von Fallbeispielen, die eine Bewertung des Einsatzes Simulationsbasierter Frühwarnsysteme ermöglichen. Diese Beispiele werden explizit beschrieben, wobei jeweils die Datenversorgung, die Simulationsmodelle, die Ausnahmedetektion und Handlungsalternativengenerierung erörtert, sowie die konkreten Ergebnisse diskutiert werden.

7.2 Kritische Würdigung

Kapitel 3 erläutert die allgemeinen Grundlagen und Zusammenhänge Simulationsbasierter Frühwarnsysteme. Dies wirft viele Fragen auf, die in den nachfolgenden Kapiteln beantwortet werden. So stellt sich etwa die Frage nach der Gestaltung der Anbindung heterogener Datenquellen und einer effizienten Datenverwaltung. In diesem Zusammenhang ist zu klären, welche Rolle XML spielt und hinsichtlich der gestellten Anforderungen an ein SEWS unterstützen kann. Es erfolgt die Erläuterung der Einbindung unterschiedlicher Simulatoren. Dabei sind besonders der notwendige Modellaufbau, in das Simulationsmodell zu implementierende Module und die Initialisierung von Simulationsmodellen von Interesse. Weiterhin stellt sich die Frage, wie Ausnahmefälle detektiert werden können. Darüber hinaus sind die Anforderungen Adaptionfähigkeit und Skalierbarkeit hinsichtlich ihrer Auswirkungen auf die Prognosegenauigkeit zu diskutieren. Ein weiteres komplexes Themengebiet stellt die Generierung von Handlungsalternativen dar. Hierzu sind die notwendigen Komponenten und Funktionalitäten zu diskutieren. Und schließlich sind sowohl die technische Realisierung und Integration in einer Unternehmensinfrastruktur als auch die praktische Umsetzung anhand von Fallbeispielen von Bedeutung.

Das Kapitel 4 geht auf die technische Umsetzung von SEWS und die Realisierung der Prognose möglicher Ausnahmefälle im Bereich von Produktion und Logistik ein. Dabei werden die Fragestellungen des vorangehenden Kapitels zur Anbindung heterogener Datenquellen und der internen Datenverwaltung Simulationsbasierter Frühwarnsysteme aufgegriffen. Weiterhin wird die Einbindung der Simulation, der hierfür notwendigen Module und die Initialisierung von Simulationsmodellen dargestellt. Auch die Prognose und Detektion von Ausnahmefällen sind diskutierte Themenschwerpunkte. Die erläuterte Vorgehensweise ermöglicht eine flexible und simulatorunabhängige Reportgenerierung. Darüber hinaus ist sie zur automatisierten Untersuchung und zeitlichen Lokalisierung interessanter Vorgänge wie bspw. Materialflusstauungen oder Kapazitätsschwankungen während der Durchführung von Simulationsexperimenten geeignet.

Die in Kapitel 5 aufgezeigte Methodik berücksichtigt sowohl eine Klassifizierung möglicher Ausnahmefälle zur Verringerung der Anzahl zu betrachtender Alternativen als auch eine adaptive Vorgehensweise zur Anpassung eines SEWS an ein spezifisches Materialflusssystem. Die verschiedenen Steuerparameter werden anhand ihrer Beeinflussungsmöglichkeit bezüglich bestimmter Ausnahmefälle ausgewählt. Dabei kommen zunächst diejenigen Steuerparameter zur Anwendung, die entsprechend des Wertes ihres Korrelationskoeffizienten die größte Erfolgswahrscheinlichkeit versprechen. Die mehrstufige Vorgehensweise ermöglicht eine Kombination verschiedener Handlungsalternativen, indem eine bisher erreichte Verbesserung mit ihrem entsprechendem Datenset (vgl. Abschnitt 5.3.5) als Eingangswerte für die Überprüfung der nächsten verwendbaren Handlungsalternative bzw. Steuerparameter berücksichtigt wird. In Abhängigkeit eines spezifischen Materialflusssystems und den zu detektierenden Ausnahmefällen, müssen entsprechende Belegungsplanungsalgorithmen bzw. -verfahren entwickelt werden. Dabei sind die zu berücksichtigenden Parameter vollständig in den SEWS- und simulationsrelevanten Daten enthalten, die als Eingangsdaten für die Simulation dienen.

Mit der prototypischen Umsetzung von SEWS befasst sich Kapitel 6. Die zuvor vorgestellten theoretischen Ansätze werden hinsichtlich ihrer Praxistauglichkeit bewertet. Die Datenbeschaffung und -versorgung wird durch die überwiegende Anzahl der auf modernen Datenbanktechnologien basierenden betrieblichen Informationssysteme erleichtert. Die verwendeten Simulatoren unterstützen die technische Umsetzung eines SEWS entsprechend der in den Abschnitten 3.3 und 3.4 vorgestellten Architektur.

7.3 Schlussfolgerungen und Ausblick

Neben vielen Antworten werden neue und weiterführende Fragen aufgeworfen. Hierzu gehören die im Zusammenhang mit den Zwischenankunftszeiten der relevanten Zustandsänderungen Z^* verbundenen Problemstellungen (vgl. Abschnitt 4.3.2.2). Dabei geht es um die Rückführung aufgetretener Ausnahmefälle auf die Verwendung minimaler Zwischenankunftszeiten und die praktische Relevanz von Ausnahmen mit sehr kurzen Grenzwertüberschreitungen. Weiterhin sind sowohl die Thematik der Entwicklung und Analyse geeigneter Messverfahren für die Prognosegenauigkeit als auch die vollständige Automatisierung der Modellaktualisierung zu diskutierende Fragestellungen für künftige Forschungsarbeiten.

Die im Kapitel 5 vorgestellte Methode zur Handlungsalternativengenerierung ist verbesserungsfähig. Es konnte grundsätzlich gezeigt werden, dass Ausnahmefälle detektiert und eine Lösungsstrategie vorgeschlagen werden kann. Komplexere Lösungsmöglichkeiten, die eine Kombination mehrerer möglicher Handlungsalternativen beinhalten, sind davon noch ausgeschlossen und führt zu interessanten Problemstellungen für nachfolgende wissenschaftliche Arbeiten. Somit ist die Handlungsalternativengenerierung weiterhin ein komplexes Themengebiet, das noch für die nächsten Jahre die Entwicklung eines standardisierten Softwarewerkzeuges für alle möglichen Materialflusssysteme zur reinen Utopie macht.

Der im Rahmen dieser Arbeit vorgeschlagene Algorithmus ist weiter zu verfeinern, damit adäquate Handlungsalternativen schnell identifiziert werden können. Dabei ist die Idee, die Anzahl möglicher Handlungsalternativen durch die Kombination einer Klassifizierung und einem Adaptionungsverfahren zu reduzieren, aus heutiger Sicht ein sinnvoller Ansatz. Darüber hinaus ist die Konfiguration des Adaptionungsverfahrens weiter zu verbessern.

Insgesamt negativ ist die gegenwärtig aufwändige Pflege und kontinuierliche Anpassung des gesamten SEWS und der verwendeten Simulationsmodelle. Sollen diese möglichst schnell für neue Materialflusssysteme einsetzbar sein, so muss eine Standardisierung der Implementierungsmethoden erfolgen und entsprechend geschultes Personal vorhanden sein. Die ohnehin hohen Anforderungen an einen Simulationsexperten werden dadurch gesteigert (vgl. Abschnitt 2.1.3). Eine effiziente und übersichtliche Benutzerschnittstelle wirkt an dieser Stelle unterstützend und erhöht die Benutzerfreundlichkeit.

Die Vorteile des Einsatzes von SEWS im Bereich von Produktion und Logistik liegen in der erfolgreichen Detektion von Ausnahmefällen, die bereits frühzeitig vermieden werden können.

Dabei lassen sich Messpunktabweichungen über den Prognosezeitraum hinweg von Schichtpausen, Schichtwechseln, spontan auftretenden Maschinenausfällen und manuellen Tätigkeiten mit evtl. manuellen Rückmeldepunkten ableiten. Diese sind analysierbar und deren Einwirkung kann abgeschätzt werden, so dass selbst bei größeren Messpunktabweichungen eine Tendenz zu möglichen Ausnahmen hin erkennbar ist. Die Potenziale liegen eindeutig auf der Hand. Lässt sich in einem Industrieunternehmen eine ausreichende Datenqualität und Datenaktualität, sowie eine geeignete und möglichst automatisierte Datenerfassung an den Messpunkten finden, nehmen die Prognosegüte und die Länge der möglichen Prognosezeiträume eines SEWS zu.

Vorteilhaft ist die frühzeitige Detektion üblicher Ausnahmefälle, die im betrieblichen Alltag häufig auftreten. Dabei liegen die Potenziale in einer schnellen und automatisierten Handlungsalternativengenerierung, die mit der Warnung vor künftigen Ausnahmefällen sofort eine geeignete Gegenmaßnahme liefert. Dabei ermöglicht die im Rahmen dieser Arbeit vorgestellte Architektur und der Einsatz moderner Webtechnologien wie Web Services, XML und XSLT eine kostengünstige Entwicklung bzw. Implementierung.

An dieser Stelle werden zusammenfassend noch einmal die wichtigsten Potenziale und Vorteile von SEWS aufgeführt:

- Die Prognose künftiger Zustände des realen Materialflusssystems ermöglicht eine frühzeitige Vermeidung bzw. Milderung von Ausnahmefällen.
- Durch die Monitoringfunktionalität können Auswirkungen bereits aufgetretener Ausnahmefälle verringert werden.
- Aus einer großen Vielzahl möglicher Handlungsalternativen werden die geeignetsten ausgewählt und mit Hilfe der Simulation bewertet.
- Die Prognosegüte, sowie die Länge des Prognosezeitraumes nimmt mit einer steigenden Datenqualität zu. Selbst bei geringerer Datenqualität ist die Tendenz zu einem Ausnahmefall hin erkennbar.
- Die Verwendung von Korrelationskoeffizienten und verschiedene Steuerparameter unterstützt die Skalierbarkeit und Adaptionfähigkeit auf unterschiedliche und neue Materialflusssysteme.
- Eine flexible Architektur und die Verwendung moderner Webtechnologien erlaubt eine einfache technische und kostengünstige Umsetzung.
- Durch den vollautomatischen Ablauf der Ausnahmedetektion und Handlungsalternativengenerierung ist eine schnelle Reaktion ohne manuelles Eingreifen möglich.

Die Beispiele aus der Praxis in Form der FSG-Montage und dem Härteofen A28 sind repräsentative Teilprozesse der Getriebeproduktion bei der DaimlerChrysler AG. Dadurch lassen sich sowohl die Anwendbarkeit Simulationsbasierter Frühwarnsysteme im betrieblichen Umfeld nachweisen als auch die noch vorhandenen Mängel aufzeigen.

Anhang A

Die einzelnen Elemente des SEWS-spezifischen XML-Schemas

In diesem Abschnitt werden sowohl die einfachen als auch die komplexen Kind-Elemente des SEWS-spezifischen XML-Schemas näher erläutert.

Hierbei dienen folgende einfachen Kind-Elemente zur Verwaltung der systempezifischen SEWS-Daten:

- `<ProductionSystemID>`
Repräsentiert eine eindeutige für das reale System gültige SEWS-interne ID. Hierbei wird auch verwaltet, ob es sich um die Daten für eine bestimmte Handlungsalternative des realen Systems handelt.
- `<SystemFile>`
Beinhaltet die Benennung der XML-Datei, die diese Daten enthält.
- `<Description>`
Hier ist eine ausführlichere Beschreibung des Systems in Langtext möglich.

Die komplexen Kind-Elemente des SEWS-spezifischen XML-Schemas werden im Folgenden einzeln näher erläutert.

Prozess-System

Die Beschreibung des Prozess-Systems eines realen Materialflusssystems wird mit Hilfe des Elementes `<ProcessSystem>` erreicht (vgl. Abb. A.1). Ein Prozess-System ist die Darstellung aller durchführbaren Prozesse unter der jeweiligen Angabe einer zeitlichen Verteilung und der hierfür notwendigen Ressourcen. Es enthält eine beliebige Anzahl von möglichen Prozessen als weitere Kind-Elemente vom Typ `<Process>`. Dieses Element beinhaltet alle notwendigen Informationen eines Prozesses. Hierzu gehört eine eindeutige *ProcessID*, eine Beschreibung des Prozesses (*Description*) und die optionale Angabe der notwendigen Ressourcen mit Hilfe einer oder mehrerer eindeutigen *ResourceID*. Weiterhin kann über eine zeitliche Verteilung und der verwendeten Zeiteinheit für diese stochastische Verteilung der zeitliche Verlauf eines Prozesses beschrieben werden. Die Verwendung von Verteilungen spielt bei der späteren Prognose eine große Rolle. Wird eine stochastische Größe in Form einer Verteilung bei den Prozesszeiten angenommen, müssen mehrere Simulationsläufe zur Prognose zukünftiger Systemzustände durchgeführt werden. Wird dagegen nur eine deterministische Größe verwendet, ist hierfür nur ein Simulationslauf notwendig.

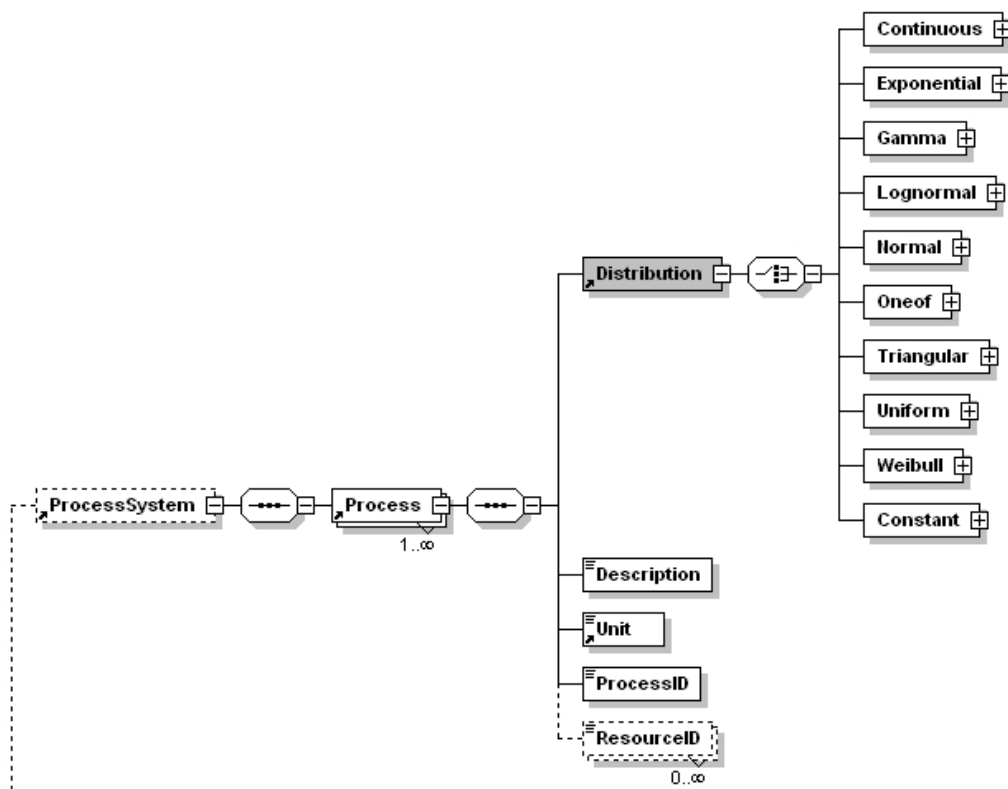


Abb. A.1: Das Kind-Element <ProcessSystem> des XML-Schemas für SEWS

Ressourcen

Die in einem realen Materialflusssystem verwendeten Ressourcen werden mit Hilfe des Elementes <Resources> verwaltet (vgl. Abb. A.2). Dabei kann es sich um beliebige Ressourcen wie Maschinen, Förderbänder, Prüfplätze oder auch Werker handeln. Jede Maschine oder Werker wird über eine eindeutige *ResourceID* und eine *Description* beschrieben. Darüber hinaus kann optional eine Fehlerverteilung unter Angabe der *Mean Time Between Failure (MTBF)* und der *Mean Time To Repair (MTTR)*, der Status der Ressource und ein Schichtmodell durch eine eindeutige *ShiftModelID* hinterlegt werden. Bei den Werkern ist es darüber hinaus möglich, die Anzahl eines bestimmten Werkertyps in einem System anzugeben. Dabei gehören zwei Werker zu dem gleichen Werkertyp, wenn diese die gleichen Arbeitsinhalte abdecken.

Messpunkte

Die in Abschnitt 4.1.1 bereits erwähnten Messpunkte eines Materialflusssystems sind zur Darstellung des aktuellen Systemzustandes von großer Bedeutung. Nur anhand dieser Messpunkte können die Positionen der einzelnen Produkte bzw. Objekte in einem Materialflusssystem identifiziert werden. Die Definition eines Messpunktes in dem SEWS-spezifischen XML-Schema erfolgt unter der Angabe einer Beschreibung (*Description*) und einer eindeutigen *Measuring-PointID* (vgl. Abb. A.3). Um einen Messpunkt im Prozess-System des Materialflusssystems zu lokalisieren, müssen Referenzprozesse angegeben werden. Unter einem Referenzprozess ist im Folgenden der Vorgänger- bzw. Nachfolgerprozess eines Messpunktes im Prozess-Systems

zu verstehen. Darüber hinaus wird eine Liste der gegenwärtig an diesem Messpunkt aktuell verbuchten Produkte mit ihrer eindeutigen *ProductID* verwaltet.

Prozess-Sequenz

Jedes Produkt ist einem bestimmten Produkttyp zugeordnet, welches eine bestimmte Prozess-Sequenz in einem Materialflusssystem durchläuft. Diese Prozess-Sequenz wird in dem XML-Schema als *ProcessSequence* bezeichnet und beinhaltet eine Beschreibung (*Description*) und eine eindeutige *ProcessSequenceID* (vgl. Abb. 5.5). Darüber hinaus kann eine Jahresstückzahl hinterlegt werden, die bspw. zur Berechnung einer Taktzeit oder von Mengenverhältnissen herangezogen werden kann. Das Element `<ProductProgram>` dient zur Beschreibung der Art des Produktionsprogramms (auch hier über eine *Description* und eine eindeutige *ProductProgramID*) wie *Losgrößenfertigung*, *Just-In-Sequence* oder *Just-In-Time*. Aus diesem Produktionsprogrammtyp ergeben sich Konsequenzen für die Steuerung der Produkte durch das Materialflusssystem. Die daraus resultierenden Steuerungsmöglichkeiten müssen bei der Modellentwicklung berücksichtigt werden. Mit Hilfe des Elements `<ProductProcess>` wird die für einen Produkttyp individuelle Prozess-Sequenz festgelegt, die eine Liste von Prozessen darstellt, die zuvor in dem Prozess-System des Kind-Elementes `<ProcessSystem>` festgelegt wurden.

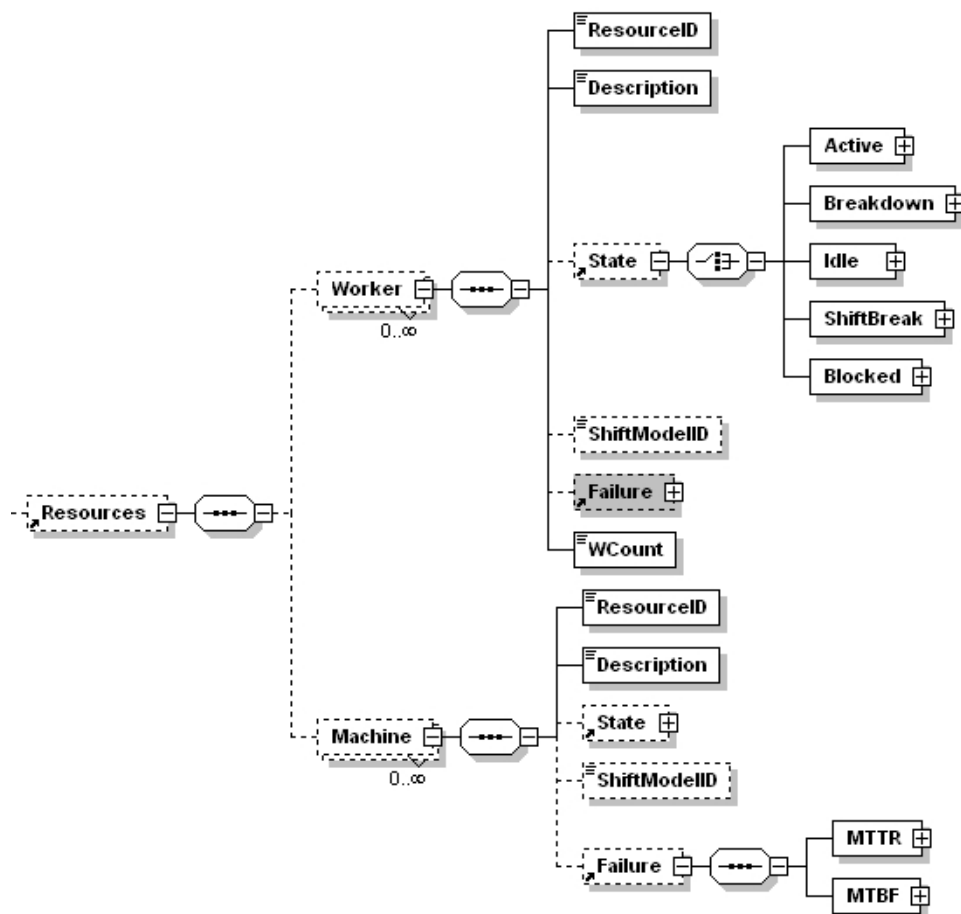


Abb. A.2: Das Kind-Element `<Resources>` des XML-Schemas für SEWS

Produkte

Die in einem Materialflusssystem gegenwärtig bearbeiteten Produkte werden mit Hilfe des Kind-Elementes `<Products>` verwaltet (vgl. Abb. A.4). Ein Produkt mit einer eindeutigen *ProductID* durchläuft ein Materialflusssystem, indem es einem oder mehreren sequentiellen Prozessen unterzogen wird. Im Zusammenhang mit Produkten ist derjenige eindeutige Messpunkt (*MeasuringPointID*) interessant, bei dem das Produkt zuletzt registriert wurde. Hierbei ist noch wichtig, zu welchem Zeitpunkt (*Timestamp*) dies geschah. Weiterhin kann die mit einem Produkt verbundene Prozess-Sequenz (*ProcessSequenceID*) und eine Beschreibung (*Description*) angegeben werden.

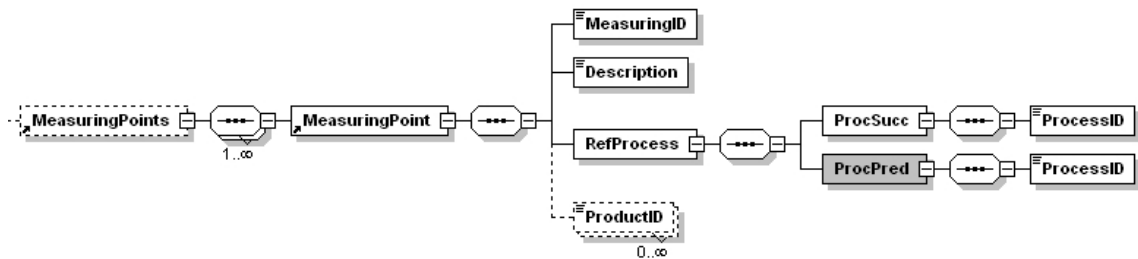


Abb. A.3: Das Kind-Element `<MeasuringPoints>` des XML-Schemas für SEWS

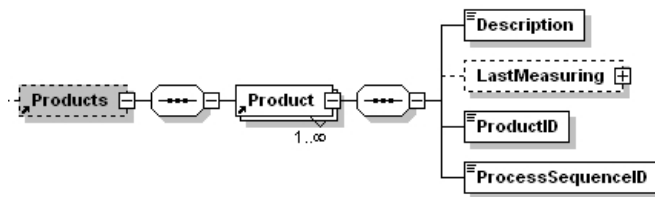


Abb. A.4: Das Kind-Element `<Products>` des XML-Schemas für SEWS

Schichtmodelle

Die Festlegung der Schichtmodelle erfolgt mit dem Kind-Element `<ShiftModels>` (vgl. Abb. A.5). Auch hier wird wieder eine eindeutige *ShiftModelID* festgelegt und es ist eine Beschreibung (*Description*) möglich. Darüber hinaus wird für jeden Wochentag die Uhrzeit des Schichtbeginns und Schichtendes und die Schichtpausen mit Beginn- und Endzeitpunkt zur Verfügung gestellt.

Produktionsplan

Um eine Prognose für einen beliebigen Prognosezeitraum mit Hilfe der Simulation erstellen zu können, muss ein gültiger Produktionsplan für diesen Zeitraum bekannt sein. Dieser Produktionsplan wird mit dem Element `<WorkOrderList>` dargestellt, das eine Liste der Aufträge unter der Angabe einer *ProductID*, einer *ProductSequenceID*, einer *Description* und eines *Timestamp* enthält (vgl. Abb. 5.6). Die Reihenfolge der Aufträge in dieser Liste entspricht deren geplanten Abarbeitung im Prognosezeitraum. Die Abbildung von Produktionsprogrammen muss aufgrund ihrer Komplexität im Simulationsmodell erfolgen. Dabei wird mit Hilfe des Zeitstempels der gewünschte Auftragsbeginn festgelegt.

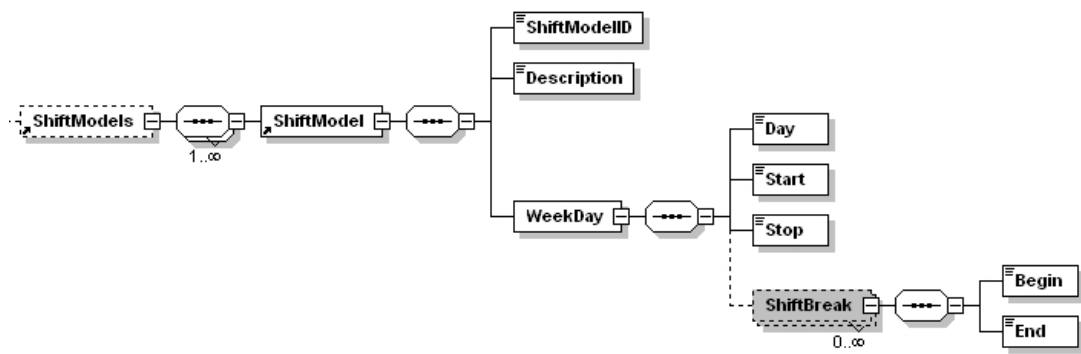


Abb. A.5: Das Kind-Element `<ShiftModels>` des XML-Schemas für SEWS

Anhang B

Performancevergleich zwischen AutoMod und SLX

Ein gutes Beispiel im Performancevergleich von Simulatoren ist die [WAIT UNTIL]-Anweisung. AutoMod verwendet eine globale Verzögerungsliste zur Verwaltung seiner Objekte, die auf die Erfüllung einer Bedingung warten müssen. Dabei werden alle Elemente (bspw. Ressourcen, Warteschlangen oder Variablen) aufgenommen, an deren Statusänderung eine Bedingungserfüllung geknüpft ist. Die zu einem aktuellen Zeitpunkt der Simulationsuhr abzuarbeitenden Ereignisse werden in einer Ereignisliste verwaltet. Wird dabei der Status mindestens eines der Elemente in der Verzögerungsliste verändert, so werden die Verzögerungsbedingungen aller Objekte in der Verzögerungsliste überprüft. Dadurch leidet die Performance der Simulationsläufe in AutoMod bei einer verstärkten Verwendung der [WAIT UNTIL]-Anweisung ([Kud00]).

SLX dagegen verwendet Kontrollvariablen zur Darstellung von Elementen, die jeweils eine eigene Verzögerungsliste besitzen. Wenn sich der Wert einer Kontrollvariable verändert, so werden nur die Objekte in dieser Verzögerungsliste auf die Erfüllung ihrer Bedingungen hin untersucht. Dadurch sind nur diejenigen Objekte betroffen, die auch tatsächlich mit dieser Kontrollvariable über eine Bedingung verknüpft sind. Aus diesem Grund ist die Verwendung von [WAIT UNTIL] in SLX viel effizienter ([Kud00]).

Simulationszeitraum	Dauer der Simulationsläufe in Sekunden			
	Variante 1		Variante 2	
	SLX	AutoMod	SLX	AutoMod
30 Tage	6,78	25,88	8,56	23,46
45 Tage	10,08	33,93	12,79	35,39
60 Tage	13,45	48,64	17,03	51,18
75 Tage	16,82	55,80	21,16	60,95
90 Tage	19,99	69,25	25,47	76,63
105 Tage	23,36	90,60	29,72	95,13
120 Tage	26,81	96,70	33,82	102,61
135 Tage	30,08	112,44	38,30	119,07
150 Tage	33,31	126,00	42,35	131,37
165 Tage	36,66	138,52	46,60	146,19
180 Tage	40,13	140,21	50,68	165,42

Tab. B.1: Performancevergleich zwischen AutoMod und SLX

Tabelle B.1 zeigt den Performancevergleich von SLX und AutoMod bei zunehmender Simulationsdauer. Hierzu wurden in AutoMod und SLX jeweils zwei identische Simulationsmodellvarianten mit den gleichen Ergebnissen erzeugt. In der einen Modellvariante werden [WAIT

UNTIL]-Anweisungen verwendet, die permanent die Veränderung des Bestandes eines Puffers in ihrer Bedingung berücksichtigen. Die zweite Modellvariante bezieht insgesamt zehn Puffer in ihre Bedingung ein. Dabei wird deutlich, dass AutoMod durchgängig die schlechtere Performance besitzt. Weiterhin nimmt die Performance mit zunehmender Simulationsdauer schneller ab als dies bei SLX der Fall ist.

Durch die konsequente Verwendung von so genannten *Order Lists* in AutoMod kann auf den Einsatz von [WAIT UNTIL]-Anweisungen verzichtet werden. Da bei den beiden Simulatoren AutoMod und SLX der Programmcode zunächst kompiliert wird und dadurch eine hohe Performance aufweisen, sind sie besonders für den Einsatz in SEWS geeignet.

Anhang C

Beispielhafte Darstellung einer Berechnungsregel

Zur Verdeutlichung der Darstellung einer Berechnungsregel mit der in Abschnitt 4.3.2.1 erläuterten Vorgehensweise, wird an dieser Stelle ein einfaches Beispiel gewählt. Seien a , b , c und d jeweils Elemente, die zur Berechnung einer Variablen x herangezogen werden, für die $x = (a + b) \cdot (c - d)$ gilt. Eine vereinfachte Darstellung dieser Berechnungsregel mit Hilfe des XML Schemas zur Definition von Kennwertberechnungen erfolgt in Abbildung C.1. Dabei kann ein Element sowohl einen absoluten Wert als auch die bereits definierte Berechnungsregel einer anderen Variablen beinhalten, wodurch die Darstellung sehr komplexer Berechnungsregeln möglich ist.

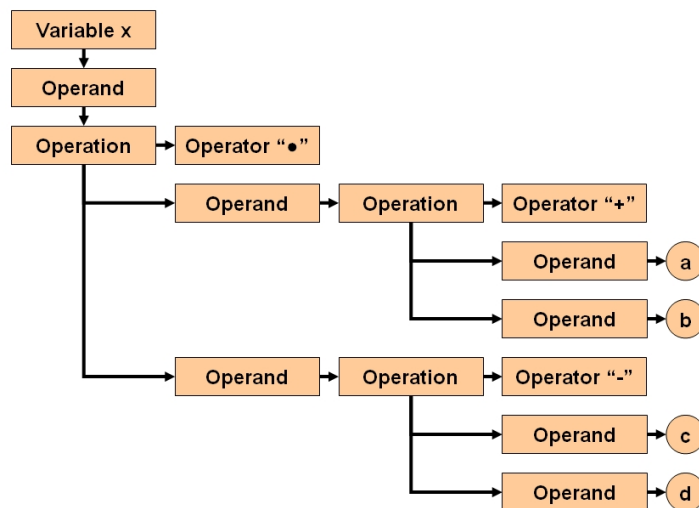


Abb. C.1: Gerichteter Baum der Variable x

Als Operatoren kommen dabei die Grundrechenarten, logische Operatoren oder Vergleichsoperatoren in Frage ([Wik07d]). Operatoren wie bspw. Verkettungszeichen für die Komposition von Funktionen und der Klassenbildungsoperator sind spezielle mathematische Operatoren, die im Folgenden außer Acht gelassen werden können. Simulatoren stellen bereits komplexe statistische Berechnungsfunktionen und Analysemöglichkeiten für Simulationsläufe zur Verfügung, die über die Attributwerte der Simulations-Entitäten verwendbar sind. Anschließend kann eine Berechnung bzw. Verknüpfung dieser Attributwerte durch die Verwendung der Operatoren zu komplexen Berechnungsregeln wie bspw. Durchschnittswerte einer Auswahl von Attributen mehrerer Simulations-Entitäten eines Typs (z.B. Werkerauslastungen) oder maximale Bestandsentwicklungen über mehrere Produkttypen hinweg erfolgen.

Abkürzungsverzeichnis

AG	Aktiengesellschaft
ASCII	American Standard Code for Information Interchange
ATF	Animation Trace Files
AVI	Audio Video Interleave
AW	Antriebswellen
B2B	Business-To-Business
BDE	Betriebsdatenerfassung
BPEL	Business Process Execution Language
BPELJ	Business Process Execution Language for Java
BPMN	Business Process Modeling Notation
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CAP	Computer Aided Planning
CAQ	Computer Aided Quality
CIM	Computer Integrated Manufacturing
COM	Component Object Model
ConWIP	Constant Work in Process
CORBA	Common Object Request Broker Architecture
CSSL	Continuous System Simulation Language
DBSM	Datenbankmanagementsystem
DCOM	Distributed Component Object Model
DDE	Dynamic Data Exchange
DSS	Decision-Support-System
DTD	Document Type Definition
DV	Datenverarbeitung
DXF	Drawing Interchange Format
EAI	Enterprise Application Integration
EDV	Elektronische Datenverarbeitung
ERP	Enterprise Resource Planning
ET	Error Total
FSG	Frontschaltgetriebe
GoA	Generator of Alternatives
GPSS	General Purpose Simulation System

HIL	Hardware-In-The-Loop
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HW	Hauptwellen
IBM	International Business Machines
ID	Identifier
IT	Informationstechnologie
IT	Informationstechnologie
JIS	Just-In-Sequence
JIT	Just-In-Time
JVM	Java Virtual Machine
KNN	Künstliche Neuronale Netze
Konti-Schicht ...	Kontinuierliche Schichtarbeit
LOD	Level of Detail
MAD	Median Absolute Deviation
MAPE	Mean Absolute Percentage Error
MathML	Mathematical Markup Language
MDE	Maschinenbezogene Datenerfassung
MDSS	Modellgesteuerte Decision-Support-Systeme
MPE	Mean Percentage Error
MSE	Mean Squared Error
MTBF	Mean Time Between Failure
MTTR	Mean Time To Repair
NC	Numeric Control
ODBC	Open DataBase Connectivity
OMG	Object Management Group
OPC	Openness, Productivity, Collaboration
PDE	Personenbezogene Datenerfassung
PPS	Produktionsplanung und -steuerung
R&S	Ranking and Selection
RMI	Remote Method Invocation
RMSE	Root Mean Square Error
RPC	Remote Procedure Call
RSM	Response Surface Methodology
SBS	Simulation-based Scheduling
SDX	Simulation Data Exchange
SEWS	Simulation-based Early Warning System
SGML	Standard Generalized Markup Language
SIMULA	Simulation Language

SLX	Simulation Language with Extensibilities
SO	Simulationsbasierte Optimierung
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SPO	Sample Path Optimization
SPS	Speicherprogrammierbare Steuerung
SQL	Structured Query Language
UDDI	Universal Description, Discovery and Integration
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VDI	Verband Deutscher Ingenieure
VW	Vorgelegewellen
W3C	World Wide Web Consortium
WMV	Windows Media Video
WS-CDL	Web Services Choreography Description Language
WSDL	Web Services Description Language
XHTML	Extensible Hypertext Markup Language
XML	Extensible Markup Language
XPath	Extensible Markup Language Path
XPDL	XML Process Definition Language
XSD	XML Schema Definition
XSL	Extensible Stylesheet Language
XSLT	Extensible Stylesheet Language Transformation
XUpdate	Extensible Markup Language Update
ZAZ	Zwischenankunftszeit

Literaturverzeichnis

- [A⁺04] ARNOLD, D. et al.: *Handbuch Logistik*. Springer Verlag, Berlin, Heidelberg, New York, 2004. 76, 77
- [Act07] ACTIVE ENDPOINTS INCORPORATION, <http://www.activebpel.org/info/intro.html>: *active BPEL*, January 2007. 53
- [ADtHW07] AALST, W. M. P. VAN DER, M. DUMAS, A. H. M. TER HOFSTEDTE und P. WOHED: *Pattern Based Analysis of BPML (and WSCI)*. Technischer Bericht, Business Process Management, <http://www.bpm.fit.qut.edu.au/about/publications/technical.jsp>, January 2007. 53
- [AF05] ARNOLD, D. und K. FURMANS: *Materialfluss in Logistiksystemen*. Springer, 4. aktualis. Aufl. Auflage, 2005. 9, 15, 23, 30, 60, 76, 77, 84, 86, 122
- [Alb05] ALBERS, S.: *Prognoserechnung*. Physica-Verl., 6. völlig neu bearb. u. erw. Aufl. Auflage, 2005. 89
- [Alt06] ALTOVA GMBH, <http://www.altova.com/de/products.html>: *Homepage Altova XMLSpy*, 2006. 64
- [And05] ANDRADÓTTIR, S.: *An Overview of Simulation Optimization via Random Search*. In: HENDERSON, S.G. und B.L. NELSON (Herausgeber): *Handbooks in Operations Research and Management*. Elsevier, 2005. 97
- [Apa07] THE APACHE SOFTWARE FOUNDATION, <http://xml.apache.org/xindice/>: *Homepage Apache Xindice*, 2007. 66
- [AWF85] AWF, AUSSCHUSS FÜR WIRTSCHAFTLICHE FERTIGUNG: *Integrierter EDV-Einsatz in der Produktion - Begriffe, Funktionszuordnungen*. Technischer Bericht, Ausschuss für wirtschaftliche Fertigung (AWF), Eschborn, 1985. 16
- [Axe03] AXEL, S.: *Dynamisches Risiko Management*. Deutscher Universitätsverlag, 2003. 29
- [Bac97] BACKES, M.: *Simulationsunterstützung zur zielorientierten Produktionsprozeßplanung und -regelung*. VDI-Verlag, 1997. iv, 10, 11, 13
- [Bal01] BALZERT, H.: *Software-Entwicklung*. Spektrum, 2. Aufl. Auflage, 2001. 39, 70
- [Ban05] BANKS, J.: *Discrete-event system simulation*. Pearson Prentice Hall, 4. ed. Auflage, 2005. 11, 12

- [Bar05] BARTON, R. R.: *Response surface methodology*. In: HENDERSON, S.G. und B.L. NELSON (Herausgeber): *Handbooks in Operations Research and Management*. Elsevier, 2005. 97
- [Bat03] BATES, C.: *XML in theory and practice*. Wiley, Chichester, 2003. 41
- [Bay03] BAYER, J.: *Simulation in der Automobilproduktion*. Springer, 2003. 14, 20, 93
- [Ber00] BERNARD, B.: *Integration of manufacturing simulation tools with information sources*. Licentiate Thesis, Royal Institute of Technology, Stockholm, Sweden, 2000. 57, 61
- [Beu06] BEUCHER, O.: *MATLAB und Simulink*. Pearson Studium, 3., überarb. Aufl. Auflage, 2006. 22
- [Bie06] BIEBERSTEIN, N.: *Service-oriented architecture (SOA) compass*. IBM Press Pearson Education, 2. print. Auflage, 2006. 74
- [BKI06] BEIERLE, C. und G. KERN-ISBERNER: *Methoden wissensbasierter Systeme*. Vieweg, 3., erw. Aufl. Auflage, 2006. 99
- [Bol04] BOL, G.: *Deskriptive Statistik*. Oldenbourg Verlag, 6. überarb. Auflage, 2004. 107
- [Bor07] BORLAND SOFTWARE CORPORATION, <http://www.borland.com/de/products/-jbuilder/index.html>: *Homepage Borland JBuilder*, 2007. 52
- [Bos04] BOSSEL, H.: *Systeme, Dynamik, Simulation*. Books on Demand, 2004. 13
- [BR93] BECKER, J. und M. ROSEMAN: *Logistik und CIM*. Springer, 1993. 16
- [BR97] BISCHOFBERGER, W. und D. RIEHLE: *Löst CORBA wirklich alle unsere Probleme?* OBJEKT spektrum, (6):36–39, November/Dezember 1997. 46
- [Bro02] BROCKHAUS: *Der Brockhaus in fünfzehn Bänden*. F.A. Brockhaus, Mannheim, 2. Auflage, 2002. 9, 32
- [Bro06] BROOKS AUTOMATION INC., <http://www.automod.de/cindex.php?site=start.html>: *Homepage Automod*, 2006. 73, 84, 120
- [Bru00] BRUNNER, D. T.: *Simulation in the Future*. In: BANKS, J. (Herausgeber): *Proceedings of the 2000 Winter Simulation Conference*. Joines, J.A. and Barton, R.R. and Kang, K. and Fishwick, P.A., 2000. 2, 24, 71
- [Bru04] BRUCKER, P.: *Scheduling Algorithms*. Springer, 4. ed. Auflage, 2004. 105
- [BSDL91] BUCHANAN-SMITH, M., S. DAVIES und R. LAMBERT: *A Guide to Famine Early Warning and Food Information Systems in the Sahel and Horn of Africa (IDS Research Reports)*. Institute of Development Studies (IDS), 1991. 27

- [BTS06] BARFUS, K., J. TOLUJEW und M. SCHENK: *Entwicklung eines modellgestützten strategischen Frühwarnsystems für logistische Netze einer verteilten Produktion*. In: *ASIM Fachtagung Simulation in Produktion und Logistik 2006*, Seiten 173–182. ASIM Fachtagung Simulation in Produktion und Logistik, SCS Publishing House, 2006. 91
- [Cal03] CALLAN, R.: *Neuronale Netze im Klartext*. Pearson Education, 2003. 99
- [Car00] CARSON, J.: *Simulation in the Future*. In: BANKS, JERRY (Herausgeber): *Proceedings of the 2000 Winter Simulation Conference*. Joines, J.A. and Barton, R.R. and Kang, K. and Fishwick, P.A., 2000. 71
- [CFF04] CARLSSON, C., M. FEDRIZZI und R. FULLÉR: *Fuzzy logic in management*. Kluwer Academic Publ., 2004. 98
- [Cha03] CHAUDHRI, A. B.: *XML data management*. Addison-Wesley, 1. print. Auflage, 2003. 66
- [CSG02] CHONG, C. S., A. I. SIVAKUMAR und R. GAY: *Design, Development and Application of an Object Oriented Simulation Toolkit for Real-Time Semiconductor Manufacturing Scheduling*. In: E. YÜCESAN, C.-H. CHEN, J.L. SNOWDON und J.M. CHARNES (Herausgeber): *Proceedings of the 2002 Winter Simulation Conference*, Seiten 1849–1856. Winter Simulation Conference, IEEE, Piscataway, N.J., 2002. 80
- [D⁺98] DEUTSCH, A. et al.: *XML-QL: A Query Language for XML*. Technischer Bericht, W3C, <http://www.w3.org/TR/NOTE-xml-ql>, 1998. 68
- [Dav98] DAVIS, W. J.: *On-Line Simulation: Need and Evolving Research Requirements*, Kapitel Chapter 13, Seiten 465–516. John Wiley & Sons, Inc., New York, Chichester, Weinheim, Brisbane, Singapore, Toronto, 1998. 36
- [DD05] DOMSCHKE, W. und A. DREXL: *Einführung in Operations Research*. Springer Verlag, Berlin, Heidelberg, New York, 6. überarb. und erw. Auflage Auflage, 2005. 8, 11, 14, 15
- [Die01] DIEZ, W.: *Automobilmarketing*. Verl. Moderne Industrie, 4., völlig überarb., aktualis. u. erw. Auflage, 2001. 14, 30
- [Dor06] DORISSON, V.: *Pandemic Influenza Planning: A Step-by-Step Guide For Businesses and Local Governments*. Outskirts Press, 2006. 27
- [Dot03] DOTZLER, B. J.: *Simulation*. In: *Ästhetische Grundbegriffe. Historisches Wörterbuch in sieben Bänden*, Seiten 509–534, Stuttgart, Weimar, 2003. Bd. V. 6
- [Dri01] DRIANKOV, D.: *Fuzzy logic techniques for autonomous vehicle navigation*. Physica-Verlag, 2001. 98
- [DSV97] DOMSCHKE, W., A. SCHOLL und S. VOSS: *Produktionsplanung*. Springer Verlag, 2., überarb. und erw. Auflage, 1997. 106

- [EE04] ECKSTEIN, R. und S. ECKSTEIN: *XML und Datenmodellierung*. dpunkt.verlag, 1. Aufl. Auflage, 2004. 64
- [Epp06] EPPLER, M. J.: *Managing information quality*. Springer, 2. ed. Auflage, 2006. 61
- [Erl04] ERL, T.: *Service-oriented architecture*. Prentice Hall PTR, 2004. 46
- [EVW05] ESPINOSA, J., J. VANDEWALLE und V. WERTZ: *Fuzzy logic, identification and predictive control*. Springer, 2005. 98
- [FAG05] FU, M. C., J. APRIL und F. W. GLOVER: *Simulation Optimization: A Review, New Developments and Applications*. In: M.E. KUHL, N.M. STEIGER, F.B. ARMSTRONG und J.A. JOINES (Herausgeber): *Proceedings of the 2005 Winter Simulation Conference*, Seiten 83–95. Winter Simulation Conference, IEEE, Piscataway, N.J., 2005. 97
- [Fel00] FELDMANN, K.: *Simulationsbasierte Planungssysteme für Organisation und Produktion*. Berlin, Springer Verlag, 2000. 17, 33, 37, 41, 42, 91
- [FFG97] FANDEL, G., P. FRANÇOIS und K.-M. GUBITZ: *PPS- und integrierte betriebliche Softwaresysteme*. Springer, 2., völlig Neubearb. und erw. Aufl. Auflage, 1997. 17
- [FI02] FRAUNHOFER-INSTITUT, MATERIALFLUSS UND LOGISTIK (IML): *SILVER: Simulationsbasierte Systeme zur Integration logistischer und verfahrenstechnischer Entscheidungsprozesse*. Arbeitsbereich 1 - Leitszenario, 2002. 97
- [Fis01] FISHMAN, G.S.: *Discrete event simulation*. Springer, New York, 2001. 11
- [Fis02] FISHWICK, P. A.: *Using XML for Simulation Modeling*. In: E. YÜCESAN, C.-H. CHEN, J.L. SNOWDON und J.M. CHARNES (Herausgeber): *Proceedings of the 2002 Winter Simulation Conference*, Seiten 616 – 622. Winter Simulation Conference, IEEE, Piscataway, N.J., 2002. 64
- [FR04] FOWLER, J. W. und O. ROSE: *Grand Challenges in Modeling and Simulation of Complex Manufacturing Systems*. SIMULATION: The Society for Modeling and Simulation International, 80(9):469–476, September 2004. 2, 24, 38, 88
- [Fu05] FU, M. J.: *Gradient Estimation*. In: HENDERSON, S.G. und B.L. NELSON (Herausgeber): *Handbooks in Operations Research and Management*. Elsevier, 2005. 97
- [Fuj00] FUJIMOTO, R. M.: *Parallel and distributed simulation systems*. Wiley, 2000. 115
- [Gac04] GACHET, A.: *Building Model-Driven Decision Support Systems with Dicoless*. VDF Hochschulverlag AG, Zürich, 2004. 92
- [GB03] GREULICH, W. und S. BARNERT: *Der Brockhaus Computer und Informationstechnologie*. Brockhaus, 2003. 34, 35, 70, 78, 106

- [GB05] GRAUPNER, T.-D. und S. BIERSCHENK: *Erfolgsfaktoren bei der Einführung der Digitalen Fabrik*. *Industrie-Management* 21, 2:59–62, 2005. 19
- [Gib00] GIBSON, R.: *Simulation in the Future*. In: BANKS, JERRY (Herausgeber): *Proceedings of the 2000 Winter Simulation Conference*. Joines, J.A. and Barton, R.R. and Kang, K. and Fishwick, P.A., 2000. 71
- [GK05] GROSSMANN, M. und H. KOSCHEK: *Unternehmensportale*. Springer, 2005. 118
- [GLJ⁺01] GAN, B. P., L. LIU, Z. JI, S. J. TURNER und W. CAI: *Managing Event Traces for a Web Front-End to a Parallel Simulation*. In: PETERS, B.A., J.S. SMITH D.J. MEDEIROS und M.W. ROHRER (Herausgeber): *Proceedings of the 2001 Winter Simulation Conference*, Seiten 637–644. Winter Simulation Conference, IEEE, Picataway, N.J., 2001. 84
- [Gor03] GORA, H. J.: *Einsatzfelder der Simulation in der Automobilindustrie*. Springer, Berlin, 2003. 8, 14
- [Gro94] GROSCH, J.: *Einsatzhärten*. expert-Verl., 1994. 127
- [GRS02] GRAUPNER, T.-D., H. RICHTER und W. SIHN: *Configuration, Simulation and Animation of Manufacturing Systems via the Internet*. In: E. YÜCESAN, C.-H. CHEN, J.L. SNOWDON und J.M. CHARNES (Herausgeber): *Proceedings of the 2002 Winter Simulation Conference*, Seiten 825–831. Winter Simulation Conference, IEEE, Picataway, N.J., 2002. 57, 61
- [GSB⁺02] GRAHAM, S., S. SIMEONOV, T. BOUBEZ, D. DAVIS, G. DANIELS, Y. NAKAMURA und R. NEYAMA: *Building Web Services with Java: Making Sense of XML, SOAP, WSDL and UDDI*. Sams Publishing, Indianapolis, 2002. 35, 46
- [GSS02] GUPTA, A. K., A. I. SIVAKUMAR und S. SARAWGI: *Shop Floor Scheduling with Simulation based Proactive Decision Support*. In: YÜCESAN, E., C.-H. CHEN, J.L. SNOWDON und J.M. CHARNES (Herausgeber): *Proceedings of the 2002 Winter Simulation Conference*, Seiten 1897–1902, 2002. 80
- [Gun95] GUNTER, D.: *Client server programming with RPC and DCE*. Que, 1995. 45
- [GW02] GLINSKY, E. und G. WAINER: *Performance Analysis of Real-Time DEVS Models*. In: E. YÜCESAN, C.-H. CHEN, J.L. SNOWDON und J.M. CHARNES (Herausgeber): *Proceedings of the 2002 Winter Simulation Conference*, Seiten 588–594. Winter Simulation Conference, IEEE, Picataway, N.J., 2002. 80
- [H⁺99] HAAS, L. et al.: *Transforming heterogeneous data with database middleware: Beyondintegration*, Band 22. 1999. 68
- [Hoo86] HOOPER, J. W.: *Strategy-related characteristics of discrete-event languages and models*, Band 46. 1986. 12
- [Hor03] HORTON, G.: *Simulation: Das virtuelle Labor*. Magdeburger Wissenschaftsjournal, 02 2003. 6, 7

- [HTRS03] HANISCH, A., J. TOLUJEW, K. RICHTER und T. SCHULZE: *Online Simulation of Pedestrian Flow in Public Buildings*. In: S. CHICK, P.J. SÁNCHEZ, D. FERRIN und D.J. MORRICE (Herausgeber): *Proceedings of the 2003 Winter Simulation Conference*, Seiten 1635–1641. Winter Simulation Conference, IEEE, Picataway, N.J., 2003. 35, 80
- [HTS05] HANISCH, A., J. TOLUJEW und T. SCHULZE: *Initialization of Online Simulation Models*. In: M.E. KUHL, N.M. STEIGER, F.B. ARMSTRONG und J.A. JOINES (Herausgeber): *Proceedings of the 2005 Winter Simulation Conference*, Seiten 1795–1803. Winter Simulation Conference, IEEE, Picataway, N.J., 2005. 80
- [HV01] HUTTUNEN, M. und B. VEHVILAINEN: *The Finnish Watershed Simulation and Forecasting System (WSFS)*. AGU Fall Meeting Abstracts, Seiten I9+, December 2001. 88
- [IBM07a] IBM, BEA SYSTEMS: *BPELJ: BPEL for Java technology*. IBM, BEA Systems, <http://www-128.ibm.com/developerworks/library/specification/ws-bpelj/>, January 2007. 55
- [IBM07b] IBM, MICROSOFT, SIEBEL SYSTEMS BEA SAP: *Business Process Execution Language for Web Services Version 1.1*. IBM, Microsoft, Siebel Systems, BEA, SAP, <http://www-128.ibm.com/developerworks/library/specification/ws-bpel/>, January 2007. 53
- [Jak91] JAKOBI, H. A.: *Akzeptanzförderung Simulationsanwendung*. In: *Simulation und Verstehen*, ASIM-Fachtagung, 1991. 15
- [JB98] JÜNEMANN, R. und A. BEYER: *Steuerung von Materialfluß- und Logistiksystemen*. Springer, 2. Aufl. Auflage, 1998. 16
- [JH06] JENSEN, S. und I. HOTZ: *Mit standardisierten Datenstrukturen zur integrativen Simulation*. In: *Simulation und Visualisierung 2006*, Seiten 89–104. SCS Publishing House e.V., 2006. iv, 68, 69, 70
- [JMS04] JURIC, M. B., B. MATHEW und P. SARANG: *Business process execution language for web services*. Packt Publ., 1. publ. Auflage, 2004. 53
- [Joh01] JOHANSSON, M.: *Information management for manufacturing system development*. Ph. D. Thesis, Division of Computer Systems for Design and Manufacturing, Department of Production Engineering, Royal Institute of Technology, Stockholm, Sweden, 2001. 61
- [Kat01] KATASTROPHENVORSORGE, ZWEITES FORUM: *Extreme Naturereignisse - Folgen, Vorsorge, Werkzeuge*. Technischer Bericht, Deutsches Komitee für Katastrophenvorsorge, <http://www.dkkv.org/DE/publications/sonderberichte.asp?h=1&MOVE=2>, September 2001. 28

- [Kel02] KELLER, W.: *Enterprise Application Integration*. dpunkt-Verl., 1. Aufl. Auflage, 2002. 45, 49
- [KG95] KOŠTURIK, J. und M. GREGOR: *Simulation von Produktionssystemen*. Springer, Wien, 1995. 12
- [Kie98] KIEF, H. B.: *FFS-Handbuch*. Hanser, 4., überarb. Aufl. Auflage, 1998. 16
- [KN03] KIM, S.-H. und B. L. NELSON: *Selecting the Best System: Theory and Methods*. In: S. CHICK, P.J. SÁNCHEZ, D. FERRIN und D.J. MORRICE (Herausgeber): *Proceedings of the 2003 Winter Simulation Conference*, Seiten 101–112. Winter Simulation Conference, IEEE, Picataway, N.J., 2003. 97
- [Knu03] KNUT, M.: *Web Services: Einführung und Übersicht*. Software & Support Verlag, Frankfurt, 2003. 47
- [KR98] KUHN, A. und M. RABE: *Simulation in Produktion und Logistik: Fallbeispielsammlung*. Springer-Verlag, Berlin, Heidelberg, New York, 1998. 8, 23, 31
- [Kud00] KUDLICH, T.: *Optimierung von Materialflusssystemen mit Hilfe der Ablaufsimulation*. Doktorarbeit, Lehrstuhl für Fördertechnik Materialfluss Logistik der Technischen Universität München, 2000. 120, 145
- [KV00] KIRCHMAIR, T. und M. VORDERWINKLER: *Soft-Commissioning: Hardware-In-The-Loop-Based Verification of Controller Software*. In: *Proceedings of the 2000 Winter Simulation Conference*, Seiten 893–899. Winter Simulation Conference, IEEE, Picataway, N.J., 2000. 36
- [LGW99] LINNER, S., M. GEYER und A. WUNSCH: *Optimierte Prozesse durch Digital Factory Tools / Optimizing Manufacturing Processes with Digital Factory Tools*, Band 1. 1999. iv, 18
- [Lie92] LIEBL, F.: *Simulation: Eine problemorientierte Einführung*. Oldenbourg, München, 1992. 12
- [Liv02] LIVINGSTON, D.: *Advanced SOAP for Web development*. Upper Saddle River, NJ: Prentice Hall PTR, 2002. 46
- [LL04] LUO, Y. und Y.T. LEE: *A Database Application for Manufacturing Simulation System Integration*. In: *Modelling and Simulation 2004 (MS 2004), 01. - 03. March 2004, Marina del Rey, CA, USA*, Seiten 161–166. International Association of Science and Technology for Development (IASTED), ACTA Press, 2004. 57
- [Loh94] LOHRBACH, T.: *Einsatz von Künstlichen Neuronalen Netzen für ausgewählte betriebswirtschaftliche Aufgabenstellungen und Vergleich mit konventionellen Lösungsverfahren*. Unitext-Verlag Göttingen, 1994. 99
- [Lov02] LOVELL, D.: *XSL Formatting Objects: Developer's Handbook*. Sams Publishing, Indianapolis, 2002. 73, 78

- [LQM03] LU, R. F., G. QIAO und C. MCLEAN: *NIST XML Simulation Interface Specification at Boeing: A Case Study*. In: S. CHICK, P.J. SÁNCHEZ, D. FERRIN und D.J. MORRICE (Herausgeber): *Proceedings of the 2003 Winter Simulation Conference*, Seiten 1230 – 1237. Winter Simulation Conference, IEEE, Picataway, N.J., 2003. 64
- [LRO96] LEVY, A., A. RAJARAMAN und J. J. ORDILLE: *Querying heterogeneous information sources using source descriptions*. Bombay, India, September 3-6 1996. 68
- [Mab00] MABROUK, K. M.: *Simulation in the Future*. In: BANKS, JERRY (Herausgeber): *Proceedings of the 2000 Winter Simulation Conference*. Joines, J.A. and Barton, R.R. and Kang, K. and Fishwick, P.A., 2000. 71
- [Mai92] MAI, W.: *CIM-Marktübersicht: Fertigungs- und Personalleitstand*. Vieweg, 1992. 17
- [MCH⁺04] MAZUR, F., R. CHROBOK, S.F. HAFSTEIN, A. POTTMEIER und M. SCHRECKENBERG: *Future of Traffic Information - Online-Simulation of a Large Scale Freeway Network*. In: *Proceedings of the IADIS International Conference WWW/Internet 2004*, Band 1, Seiten 665–672, 2004. 80
- [McL02] MCLAUGHLIN, B.: *Java and XML data binding*. O'Reilly, Beijing, Cambridge, Farnham, Köln, Paris, Sebastopol, Taipei, Tokyo, 2002. 41
- [Möh99] MÖHR, W.: *SGML und XML*. Springer, 1999. 41
- [MJLR02] MCLEAN, C., A. JONES, T. LEE und F. RIDDICK: *An Architecture for a Generic Data-Driven Machine Shop Simulator*. In: E. YÜCESAN, C.-H. CHEN, J.L. SNOWDON und J.M. CHARNES (Herausgeber): *Proceedings of the 2002 Winter Simulation Conference*, Seiten 1108 – 1116. Winter Simulation Conference, IEEE, Picataway, N.J., 2002. 64
- [MSX00] MILLER, J. A., A. F. SEILA und X. XIANG: *The JSIM Web-Based Simulation Environment*. Future Generation Computer Systems (FGCS), Special Issue on Web-Based Modeling and Simulation, Elsevier North-Holland, 17(2):119–133, October 2000. 22
- [MvNV⁺99] MAASSEN, J., R. VAN NIEUWPOORT, R. VELDEMA, H. E. BAL und A. PLAAT: *An efficient implementation of Java's remote method invocation*. In: *PPoPP '99: Proceedings of the seventh ACM SIGPLAN symposium on Principles and practice of parallel programming*, Seiten 173–182, New York, NY, USA, 1999. ACM Press. 45
- [MW03] MEIER, A. und T. WÜST: *Objektorientierte und objektrelationale Datenbanken*. dpunkt-Verl., 3., überarb. u. aktualisierte Aufl. Auflage, 2003. 66
- [NM02] NEUMANN, K. und M. MORLOCK: *Operations Research*. Hanser, 2. Aufl. Auflage, 2002. 38, 82, 95, 97

- [NRS04] NEUBAUER, B., T. RITTER und F. STOINSKI: *CORBA-Komponenten*. Springer, 2004. 45
- [Oas06] OASIS: *WS-BPEL 2.0 Committee Draft*. Technischer Bericht, Organization for the Advancement of Structured Information Standards, http://www.oasis-open.org/committees/document.php?document_id=18714&wg_abbrev=wsbpel, May 2006. 53
- [Obe99] OBERWEIS, A.: *Software-Management '99*. Teubner, 1999. iv, 7
- [oEW06] EARLY WARNING, EWC III THIRD INTERNATIONAL CONFERENCE ON: *Developing Early Warning Systems: A Checklist*. Technischer Bericht, UN Inter-Agency Secretariat of the International Strategy for Disaster Reduction (UN/ISDR), http://www.ewc3.org/upload/downloads/checklist.final_pdf.pdf, March 2006. 28
- [OMG07] OMG: *Business Process Modeling Notation Specification*. Technischer Bericht, The Object Management Group, <http://www.omg.org/docs/dtc/06-02-01.pdf>, January 2007. 53
- [Ora07] ORACLE: *Oracle BPEL Process Manager*. Oracle, <http://www.oracle.com/technology/products/ias/bpel/index.html>, January 2007. 53
- [Pag91] PAGE, B.: *Diskrete Simulation: eine Einführung mit Modula-II*. Springer, Berlin, 1991. 10, 11
- [PGN06] PERSSON, A., H. GRIMM und A. NG: *On-Line Instrumentation for Simulation-Based Optimization*. In: L.F. PERRONE, F.P. WIELAND, J. LIU B.G. LAWSON D.M. NICOL und R.M. FUJIMOTO (Herausgeber): *Proceedings of the 2006 Winter Simulation Conference*, Seiten 304–311. Winter Simulation Conference, IEEE, Picataway, N.J., 2006. iv, 35, 36, 97, 98
- [PP04] POMBERGER, G. und W. PREE: *Software engineering*. Hanser, 3., vollst. überarb. Aufl. Auflage, 2004. 23
- [RB01] RANDELL, L. G. und G. S. BOLMSJÖ: *Database Driven Factory Simulation: A Proof-of-Concept Demonstrator*. In: PETERS, B.A., J.S. SMITH D.J. MEDEIROS und M.W. ROHRER (Herausgeber): *Proceedings of the 2001 Winter Simulation Conference*, Seiten 977–983. Winter Simulation Conference, IEEE, Picataway, N.J., 2001. 57, 68
- [Rea05] READ, M.: *Konzeption und Inbetriebnahme eines Frühwarnsystems*. Technischer Bericht, comration GmbH Technology & Consulting, Zürich, April 2005. 29
- [Rib00] RIBBENS, J. A.: *Simultaneous Engineering for New Product Development*. John Wiley & Sons, New York, 2000. 18

- [RJ03] REINHARDT, A. und S. JENSEN: *Integration industrieller DV-Systeme zur automatischen Modellgenerierung in der Getriebeproduktion*. In: SCHULZE, T., S. SCHLECHTWEG und V. HINZ (Herausgeber): *Simulation und Visualisierung 2003*. SCS European Publishing House, Erlangen, 2003. 42
- [RN03] RUSSELL, S. und P. NORVIG: *Artificial Intelligence: A Modern Approach*. Pearson Education Inc., Upper Saddle River, New Jersey, 2003. 99
- [RNJ03] REINHARDT, A., V. NEMRUDE und S. JENSEN: *Formale Beschreibung von Simulationsmodellen in XML*. In: HOHMANN, R. (Herausgeber): *Simulationstechnik - 17. Symposium in Magdeburg*, Seiten 69–74. Erlangen, 2003, SCS-Europe, 2003. iv, 41, 58
- [Sap06] SAPHIRO, R. M.: *XPDL 2.0: Integrating Process Interchange and BPMN*, Kapitel 2, Seiten 183–204. Future Strategies, 2006. 53
- [SB06] SCHRIBER, T. J. und D. T. BRUNNER: *Inside Discrete-Event Simulation Software: How It Works and Why It Matters*. In: L.F. PERRONE, F.P. WIELAND, J. LIU B.G. LAWSON D.M. NICOL und R.M. FUJIMOTO (Herausgeber): *Proceedings of the 2006 Winter Simulation Conference*, Seiten 118 – 128. Winter Simulation Conference, IEEE, Picataway, N.J., 2006. 73
- [Sch99] SCHLENDER, D.: *Anforderungen an die Detailstufenverwaltung in virtuellen Simulationsumgebungen*. VDI-Verl., 1999. 72
- [Sch01] SCHNEIDER, S.: *Rechnergestützte, kooperativ arbeitende Optimierungsverfahren am Beispiel der Fabriksimulation*. 2001. 13
- [Sch02] SCHILLER, E.: *Digitale Fabrik*, Band 2. 2002. 18
- [Sch03] SCHUMANN, M.: *Implementierung von verteiltem Training unter Nutzung der High Level Architecture*. SCS-Europe, Erlangen, 2003. 13
- [SH98] SCHULZE, T. und J. HENRIKSEN: *Simulation Needs SLX: Handbuch zum Simulationssystem SLX*. Technischer Bericht, Otto-von-Guericke Universität Magdeburg, Fakultät für Informatik, 1998. 73, 120, 129
- [Sku04] SKULSCHUS, M.: *XML Schema - Vollständige Einführung, Grundlagen, Praxis, Referenzen*. Galileo Press, Bonn, 2004. 41, 64
- [SPJS99] SMITH, J. S., B. A. PETERS, S. E. JORDAN und M. K. SNELL: *Distributed real-time simulation for intruder detection system analysis*. In: *WSC '99: Proceedings of the 31st conference on Winter simulation*, Seiten 1168–1173, New York, NY, USA, 1999. ACM Press. 37
- [SSSI05] SCHMITT, J., M. SCHORN, U. STÄHLIN und R. ISERMANN: *Entwicklungs-umgebung mit echtzeitfähigen Gesamtfahrzeugmodellen für sicherheitskritische Fahrerassistenzsysteme*. *Automatisierungstechnik*, 53(1):28–35, January 2005. 36
- [Sta73] STACHOWIAK, H.: *Allgemeine Modelltheorie*. Springer, 1973. 6

- [Sta02] STAAB, S.: *Wissensmanagement mit Ontologien und Metadaten*. 2002. 61
- [Sta07] STARK, T.: *Java EE 5*. Addison Wesley, 2007. 52
- [Ste03] STEUDE, V.: *PPS-Systeme auf Basis von Simulation*. Eul, 1. Aufl. Auflage, 2003. 17
- [STK02] SNELL, J., D. TIDWELL und P. KULCHENKO: *Programming Web services with SOAP*. O'Reilly, 1. ed. Auflage, 2002. 46
- [TAL05] TURBAN, E., J. E. ARONSON und T. P. LIANG: *Decision Support Systems and Intelligent Systems*. New Jersey, Pearson Education, Inc., New Jersey, 2005. 92
- [Tha00] THALHEIM, B.: *Entity relationship modeling*. Springer, 2000. 82
- [TP07] TWISTER-PROJECT: *Twister 0.3 - Open Source WS-BPEL Engine*. java.net - The Source for Java Technology Collaboration, <http://today.java.net/pub/n/Twister0.3>, January 2007. 53
- [TvS03] TANENBAUM, A. S. und M. VAN STEEN: *Verteilte Systeme: Grundlagen und Paradigmen*. Pearson Studium, 1. Aufl. Auflage, 2003. 44, 45
- [VDI00] VDI: *Materialfluss und Fördertechnik Band 8*. VDI-Handbuch. VDI-Gesellschaft Fördertechnik Materialfluss Logistik, Düsseldorf, 2000. iv, 7, 9, 14, 19, 20, 21, 23, 37
- [VDI02] VDI: *Die digitale Fabrik wird Chefsache im Automobilbau*. VDI Nachrichten, 28:9, 2002. 19
- [VDI06] VDI: *Digitale Fabrik - Grundlagen*. VDI-Handbuch. VDI-Gesellschaft Fördertechnik Materialfluss Logistik, Düsseldorf, 2006. 18
- [Vor03] VORDERWINKLER, M.: *Betreibersimulation - Methoden für einen breiten Simulationseinsatz*. In: VORDERWINKLER, M. (Herausgeber): *Simulation in Produktion und Logistik*. Steyr, Österreich, 2003. 37
- [W3C06] W3C: *XML*. <http://www.w3.org>, January 2006. 40
- [W3C07a] W3C: *MathML*. Technischer Bericht, World Wide Web Consortium, <http://w3.org/TR/2003/REC-MathML2-20031021>, January 2007. 82
- [W3C07b] W3C: *Web Services Choreography Description Language Version 1.0*. Technischer Bericht, World Wide Web Consortium, <http://www.w3.org/TR/2005/CR-ws-cdl-10-20051109/>, January 2007. 53
- [W3C07c] W3C: *Web Services Description Language (WSDL) 1.1*. Technischer Bericht, World Wide Web Consortium, <http://www.w3.org/TR/wsdl.html>, January 2007. 54
- [Wal01] WALKER, S.: *Mechanisms to Promote User Satisfaction to Achieve Recognition of The Value of The Meteorological Services*. World Meteorological Organization - Technical Library - Online, 2001. 88

- [Wei06] WEIGERT, G.: *Theorie und Praxis der simulationsgestützten Ablaufplanung*. In: *Simulation in Produktion und Logistik 2006*, Seiten 253–262. ASIM - Arbeitsgemeinschaft Simulation, Sigrid Wenzel, 2006. 37
- [Wie05] WIEDEMANN, T.: *Simsolution - An Open Simulation Environment Founded on Extreme Multitasking*. In: M.E. KUHLE, N.M. STEIGER, F.B. ARMSTRONG und J.A. JOINES (Herausgeber): *Proceedings of the 2005 Winter Simulation Conference*, Seiten 631 – 636. Winter Simulation Conference, IEEE, Piscataway, N.J., 2005. 64
- [Wik07a] WIKIPEDIA: *Zitat Wikipedia* (http://de.wikipedia.org/wiki/Computer_Integrated_Manufacturing). Internet, 2007. 16
- [Wik07b] WIKIPEDIA: *Zitat Wikipedia* (http://de.wikipedia.org/wiki/Frühwarnung_bei_Naturkatastrophen). Internet, 2007. 28
- [Wik07c] WIKIPEDIA: *Zitat Wikipedia* (<http://de.wikipedia.org/wiki/Grasland-Feuerindex>). Internet, 2007. 27, 28
- [Wik07d] WIKIPEDIA: *Zitat Wikipedia* (http://de.wikipedia.org/wiki/Operator_Mathematik). Internet, 2007. 147
- [Wik07e] WIKIPEDIA: *Zitat Wikipedia* (<http://de.wikipedia.org/wiki/Potential>). Internet, 2007. 32
- [Wik07f] WIKIPEDIA: *Zitat Wikipedia* (<http://de.wikipedia.org/wiki/Variabilität>). Internet, 2007. 23
- [Wik07g] WIKIPEDIA: *Zitat Wikipedia* (http://en.wikipedia.org/wiki/Computer_simulation). Internet, 2007. 7
- [Wik07h] WIKIPEDIA: *Zitat Wikipedia* (<http://en.wikipedia.org/wiki/Engine>). Internet, 2007. 68
- [WL03] WILDE, E. und D. LOWE: *Xpath, XLink, XPointer, and XML*. Addison-Wesley, 2003. 66, 67
- [Wol06] WOLVERINE, <http://www.wolverine.com>: *Homepage SLX*, 2006. 73, 120
- [Wor02a] WORTMANN, D.: *Aktueller Stand und Perspektiven der Logistiksimulation*. In: PRADEL, U.-H. und W. SÜSSENGUTH (Herausgeber): *Praxishandbuch Logistik*. Loseblatt, Grundwerk Köln 2001, 2002. 37
- [Wor02b] WORTMANN, D.: *Auf dem Weg zur Traumfabrik*. Band 52, Seite 111f, 2002. 15