

Produktionsunterstützung für den Schiffbau unter Integration von Konstruktions- und Planungsdaten

Von der Fakultät für Maschinenwesen
der Rheinisch-Westfälischen Technischen Hochschule Aachen
zur Erlangung des akademischen Grades
eines Doktors der Ingenieurwissenschaften
genehmigte Dissertation

vorgelegt von

Martin Schulten

aus Krefeld

Berichter: Univ.-Prof. Dr.-Ing. P. Drews
Univ.-Prof. Dr. techn. Prof. h.c. (RC) E. Lugscheider

Tag der mündlichen Prüfung: 24 April 2003

Diese Dissertation ist auf den Internetseiten der Hochschulbibliothek online verfügbar.

Vorwort

Die vorliegende Arbeit entstand im Rahmen meiner Tätigkeit als Projektleiter am Europäischen Centrum für Mechatronik / APS GmbH in Aachen.

Herrn Prof. Dr.-Ing. P. Drews danke ich für die Anregung zu dieser Arbeit, für die Unterstützung und Förderung während der Forschungstätigkeiten sowie für die Übernahme des Referats.

Mein Dank gilt weiterhin Herrn Univ.-Prof. Dr. techn. Prof. h.c. (RC) E. Lugscheider, Leiter des Lehr- und Forschungsgebiets Werkstoffwissenschaften der RWTH Aachen, für die Übernahme des Korreferats. Herrn Prof. Dr.-Ing. J. Betten, Leiter des Lehr- und Forschungsgebiet Mathematische Modelle in der Werkstoffkunde der RWTH Aachen, danke ich für die Übernahme des Prüfungsvorsitzes.

Ferner danke ich allen Kollegen und studentischen Mitarbeitern, die mit engagiertem Einsatz zum Gelingen dieser Arbeit beigetragen haben.

Mein besonderer Dank gilt Herrn Dr.-Ing. P. Fromm und Frau Silke Schankweiler für die kritische Durchsicht des Manuskripts.

Ich danke meiner Familie, insbesondere meinen Eltern, die mir diesen Weg ermöglicht haben. Meiner Freundin Britta möchte ich herzlich für ihre Geduld und Unterstützung danken.

Köln, im Mai 2003

Martin Schulzen

Inhalt

1 Einleitung	1
1.1 Motivation und Problemstellung	1
1.2 Zielsetzung und Beitrag der Arbeit	4
2 Fertigungsorganisation im Schiffbau	7
2.1 Traditionelle Organisationsform im Schiffbau	7
2.1.1 Auftragsmanagement und -bearbeitung	7
2.1.2 Beispiel Rohrwerkstatt	12
2.1.3 Beispiel Montagewerkstatt für Strukturelemente	18
2.2 Soziale Aspekte und Kommunikation	22
2.3 Heutige prozessunterstützende Systeme	25
2.3.1 Rohrwerkstatt	27
2.3.2 Montagewerkstatt für Strukturelemente	28
3 Grundlagen der Kommunikation	29
3.1 Dezentrale Organisation	29
3.1.1 Holonisches Modell	31
3.1.2 Übertragung auf den Schiffbau	32
3.2 Kommunikation	35
3.2.1 Horizontale Kommunikation	35
3.2.2 Vertikale Kommunikation	36
3.3 Systemanforderungen	36
3.3.1 Inhaltliche Anforderungen	37
3.3.2 Technische Anforderungen	38
3.3.3 Ergonomische Anforderungen	39
3.3.4 Ökonomische Anforderungen	47
4 Technologische Plattform	49
4.1 Betriebssysteme	49
4.1.1 Client	51
4.1.2 Server	52

4.1.3 Echtzeitsystem	52
4.2 Interface-Technologien und Datenformate	53
4.3 Intranet	53
4.4 Aktive Module	54
4.4.1 Java	54
4.4.2 Tool Command Language / Toolkit (TCL/Tk)	55
4.4.3 3D-Visualisierung und modellorientierte Prozessüberwachung	56
4.5 Integrationsplattform	58
4.5.1 Middleware für verteilte Systeme	58
4.5.2 Generische Middleware zur Datenintegration	60
5 Systemarchitektur	65
5.1 Anwendungsfälle	65
5.1.1 Information	65
5.1.2 Werkerkommunikation	66
5.1.3 Prozessintegration und Verifikation	67
5.1.4 Aufbereitung von Technologiedaten	68
5.1.5 Produktionsplanung	68
5.1.6 Bewertung der Produktion	68
5.2 Systemarchitektur	69
5.2.1 Funktionsorientiertes Ebenenmodell	70
5.2.2 Teilmodule	77
6 Applikation und Integration	83
6.1 Modulares Gesamtsystem	83
6.2 Datentechnische Infrastruktur	85
6.2.1 Serversystem	85
6.2.2 Client	85
6.3 Systemkomponenten	89
6.3.1 Echtzeitfähige Integration des Fertigungsprozesses	89
6.3.2 Interaktive Simulation und Visualisierung	104
6.3.3 Integration der Fertigungsplanung	115
6.3.4 Prozessanalyse und Evaluierung	120
6.3.5 Kommunikation	123

6.3.6 Informationssystem	124
7 Einsatz und Validierung des Systems	127
7.1 Einsatz in den Werkstätten	127
7.1.1 Realisierungsbeispiel A: Rohrwerkstatt	127
7.1.2 Realisierungsbeispiel B: Montagewerkstatt für Strukturelemente	128
7.2 Akzeptanz des Systems	129
7.2.1 Akzeptanz durch die Unternehmensleitung	129
7.2.2 Benutzerakzeptanz	129
7.2.3 Gewerkschaften und Arbeiterorganisationen	131
7.3 Wirtschaftliche Aspekte	131
8 Zusammenfassung und Ausblick	133
9 Literatur	137

1 Einleitung

1.1 Motivation und Problemstellung

Die wachsende Globalisierung der Märkte führt zu einer stetigen Wettbewerbsverschärfung für die europäische Industrie. Besonders durch arbeitsintensive Fertigung geprägte Industrien, wie der Schiffbau, sehen sich starkem Druck der asiatischen Konkurrenz ausgesetzt. Während sich der Weltmarktanteil des asiatischen Schiffbaus von 62,0% im Jahr 1996 auf 65,5% in 1999 steigerte, fiel der Anteil in Deutschland hergestellter Schiffe im selben Zeitraum von 6,7% auf 4,3% zurück (Abbildung 1-1).

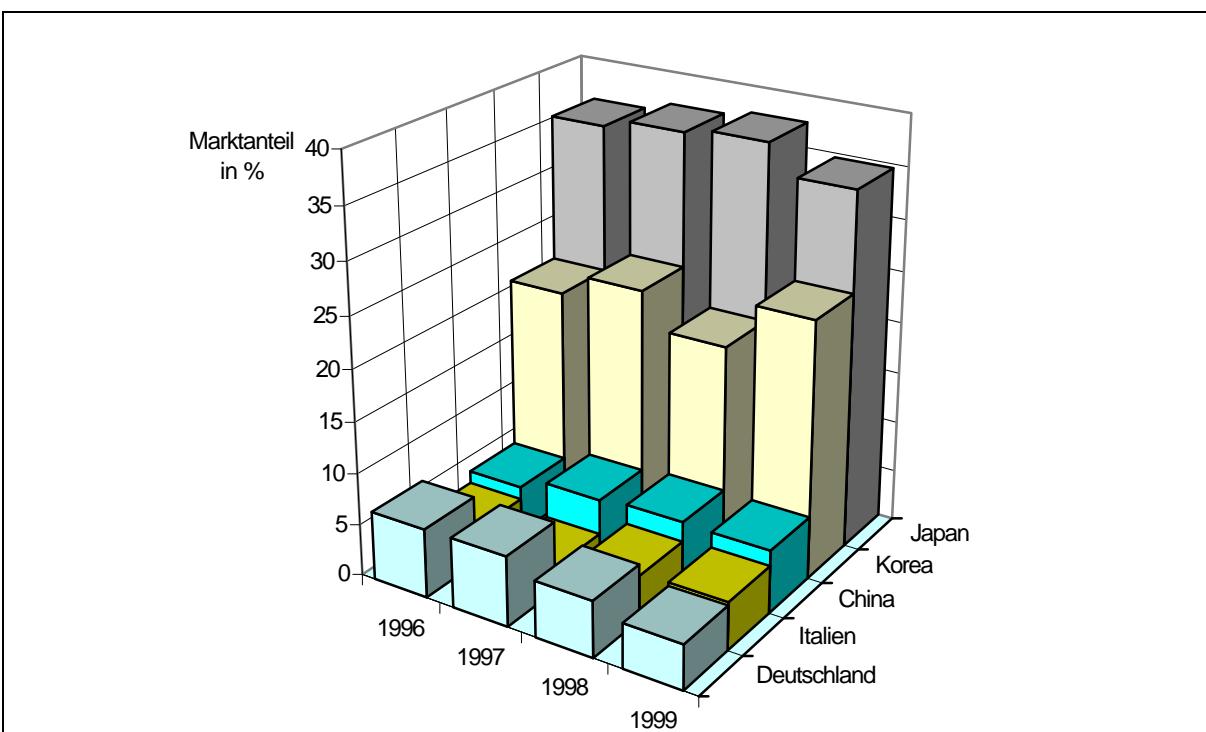


Abbildung 1-1: Marktanteile im Schiffbau [VSM00]

Dies entspricht einem relativem Rückgang des deutschen Marktanteils um 35,8%, während die asiatischen Schiffbauer einen Zuwachs von 5,6% verzeichneten.

Deutschland fiel damit von Platz drei auf Platz fünf der Statistik zurück. Die steigende Produktkomplexität, die stetige Forderung nach Innovationen sowie gleichzeitig fallende Preise forcieren zudem eine Wandlung der Kostenstruktur industrieller Unternehmen: Der globale Preiskampf bei gleichzeitig steigenden Marketing- und Entwicklungskosten erhöht den Kostendruck auf Produktion und Fertigung. Ein Containerschiff mit einer Kapazität von 3500 Standardcontainern kostete 1993 noch 70 Mio. US-Dollar. Im Jahr 1998 betrug der Preis für ein solches Schiff nur noch 42 Mio. US-Dollar, obwohl die technische Ausstattung bedingt durch internationale Sicherheitsauflagen komplexer wurde. Zur Erhaltung der Konkurrenzfähigkeit ist daher eine Steigerung der Fertigungseffizienz unabdingbar.

Meist wird versucht, die notwendige Effizienzsteigerung durch eine stärkere Automatisierung der Produktion sowie eine Straffung der Arbeitsorganisation zu erreichen. Neben dem Verlust von Arbeitsplätzen resultiert dieser Rationalisierungsweg in einer Reduktion der Eigenverantwortlichkeit der Arbeiter und vor allem im Bereich der im Schiffbau üblichen Einzelteil- und Kleinserienfertigung zur Abnahme der Flexibilität der Produktionsstätten.

Dem gegenüber steht die fortschreitende, durch Innovationen und internationalen Konkurrenzdruck bedingte Verkürzung der Produktlebenszyklen und die daraus resultierende Forderung nach einer kurzen Reaktionszeit des Unternehmens auf Forderungen des Marktes nach neuen, innovativen Produkten (Time-to-Market). Das häufig angeführte Spannungsdreieck aus Qualität, Zeit und Kosten [FIS96] muss daher um den Wettbewerbsfaktor *Innovation* ergänzt werden (Abbildung 1-2).

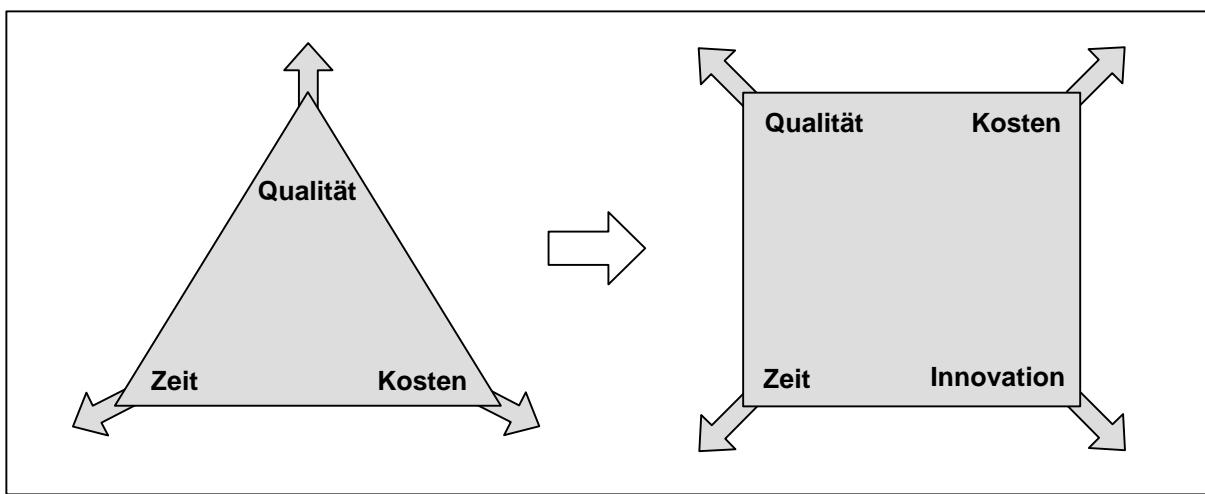


Abbildung 1-2: Wandlung des Spannungsfeldes „Qualität-Zeit-Kosten“

Innovative Produkte erhöhen meist die Produktionskosten, erfordern zeitlichen Mehraufwand und können unter Umständen zu Lasten der Qualität gehen, wenn noch zu wenige Erfahrungen mit dem neuen Produkt vorliegen. Innovation stellt demnach ein eigenständiges Optimierungskriterium dar.

Innovative Produkte erfordern neue Paradigmen in der Fertigung, da sie die Produktvarianz erhöhen und damit die Stückzahlen identischer Teilprodukte senken. Es hat sich gezeigt, dass hochautomatisierte Produktionszellen in traditionellen Top-Down-Hierarchien zwar in Teilbereichen, wie der Fertigung von hohen Stückzahlen gleicher oder sehr ähnlicher Teile, eine Steigerung der Produktionseffizienz erzielen können. In Bezug auf geringe und mittlere Losgrößen ist eine solche Lösung jedoch nicht ausreichend flexibel, um auf dem Weltmarkt mit innovativen Produkten konkurrenzfähig zu bleiben. Die Integration des Menschen, die Förderung seiner Fähigkeiten und Erfahrungen muss in heutigen Fertigungsumgebungen stärker berücksichtigt werden [MAC97].

Die straffen Hierarchien lassen dem Werker nur wenig Raum für Entscheidungen und Eigenverantwortung. Sein Potenzial, seine langjährige Erfahrung in der Produktion und sein zielgerichtetes Problemlöseverhalten vor Ort werden nicht konsequent genutzt. Motivation und Eigenverantwortlichkeit lassen nach und senken nachhaltig die Effizienz der gesamten Produktionsstätte [VER98].

Einen vielversprechenden Ansatz zur Effizienzsteigerung industrieller Werkstätten sieht man heute in der Symbiose aus Automatisierung und struktureller Reorganisation zur Förderung der menschlichen Potenziale [SCH98], [SCR98], [TLM98].

Ziel der strukturellen Reorganisation ist die Ablösung der traditionellen Top-Down-Hierarchien mit starren Fertigungslinien durch eine offene Teamstruktur mit Fertigungsinseln. Die Teamstruktur begünstigt die Auslastung von Ressourcen und steigert die Flexibilität der Fertigung und Motivation der Werker. Teammitglieder sollen in der Lage sein, weitestgehend selbstständig Entscheidungen zu treffen und Planungsaufgaben zu übernehmen. Das Potenzial der Werker, ihr Erfahrungs- und Spezialwissen wird dadurch stärker genutzt und in die Fertigung eingebracht [KIG98].

Den Werkern werden mehr Entscheidungskompetenzen eingeräumt, die eine umfassende Information über Auftragslage, Termine und Fertigungsschnittstellen erforderlich machen, wie sie in heutigen Werkstätten im Schiffbau nicht üblich ist.

Auch in anderen, ähnlich strukturierten Bereichen industrieller Produktion, wie zum Beispiel im Flugzeug- oder Waggonbau, sowie in der Fertigung von Großanlagen kann die Einführung von modernen Teamstrukturen als ein wichtiger Faktor zur Stärkung der Wettbewerbsposition auf dem Weltmarkt gesehen werden [PDT99].

1.2 Zielsetzung und Beitrag der Arbeit

Ziel einer effizienzsteigernden Maßnahme muss damit sein, bestehende, starre Hierarchien in Teilgebieten aufzulösen, um die Verantwortlichkeit und Motivation der Arbeiter sowie die Flexibilität der Produktion zu erhöhen [FLG98]. Die strukturelle Reorganisation der Fertigungsprozesse mit der Einführung von Arbeitsgruppen lässt sich nicht ohne begleitende, die neue Organisationsform unterstützende Maßnahmen verwirklichen [BAC96], [SCH98], [RYA98].

Eine partizipative Teamstruktur soll die Fertigungseffizienz steigern. Folgende Aspekte sind wesentliche Bestandteile der Einführung einer Teamstruktur:

- ➔ Abbau von starren Hierarchien
- ➔ Einführung kapazitäts- und potenzialorientierter Arbeitsweisen
- ➔ Weitreichende Autonomie der Teams

Zur Durchsetzung dieser Ziele ist die Berücksichtigung einer Vielzahl von Faktoren, die neben der rein technischen Sichtweise auch die sozialen Komponenten betreffen, notwendig. Die wichtigsten Faktoren im Schiffbau sind nach [VER98]:

- ➔ Unterstützung der Teams und Teammitglieder bei der Entscheidungsfindung durch Einführung
 - eines anwenderorientierten Informationssystems und Integration, Visualisierung und Bewertung von Prozessdaten und
 - eines themenbasierten, werkstattübergreifenden Kommunikationssystems
- ➔ Unterstützung der Arbeitsorganisation durch Integration eines
 - offenen, interaktiven Planungssystems und
 - eines Systems zur Simulation und Bewertung von Werkstattplänen
- ➔ Integration der verschiedenen Funktionalitäten in ein einfach zu bedienendes, nutzerorientiertes System

Für die Bereiche Information, Arbeitsplanung und Problemanalyse bis hin zur Prozessevaluierung sind in dieser Arbeit Systeme entwickelt worden, die dem Arbeiter innerhalb seiner Umgebung die notwendige Entscheidungsfähigkeit ermöglichen.

Um unterschiedliche Qualifikationen und Fertigkeiten optimal zu fördern, wurden diese Systeme benutzerorientiert ausgelegt und unter Verwendung moderner Schnittstellentechnologien implementiert. Die neue Organisationsform der verteilten Produktion soll optimal durch die entwickelten Systeme unterstützt werden.

Das Unterstützungssystem basiert auf einer modularen Systemarchitektur, die mithilfe moderner Intranettechnologien sowie unter Einsatz standardisierter Middleware ein hohes Integrationspotenzial in industriellen Umgebungen bietet. Die Entwicklungen umfassen unter anderem Module zur echtzeitfähigen Integration von Prozessdaten, Module zur Simulation des Fertigungsprozesses sowie zur Visualisierung von Werkstücken. Weitere Kernfunktionalitäten ermöglichen neue Kommunikationsformen, wie z.B. Notizen über einem bestimmten Werkstücktypus, Erfahrungsaustausch und den Zugriff auf Datenbanken mit allgemeinen und prozessspezifischen Informationen. Das System wird durch die Integration eines werkstattorientierten Planungssystems ergänzt.

Basis der Entwicklungen sind die Analysen der Arbeitsorganisation in zwei verschiedenen Werkstätten der Odense Steel Shipyard Ltd., einer großen dänischen Werft für Containerschiffe und Rohöltanker. Diese in Kapitel 2 dargestellten Analysen umfassen die Werkstattplanung, Fertigung, soziale Aspekte sowie bestehende Kommunikationsformen in den Werkstätten. Den Abschluss der Analysen bildet die Beschreibung der heute dort existierenden prozessunterstützenden IT-Systeme.

Kapitel 3 befasst sich mit den theoretischen Grundlagen der dezentralen Fertigungsorganisation und wendet diese auf den Schiffbau an. Es werden erforderliche Kommunikationswege aufgezeigt sowie die Systemanforderungen unter verschiedenen Gesichtspunkten definiert.

Anschließend werden in Kapitel 4 die für das Prozessunterstützungssystem relevanten Basistechnologien wie z.B. Betriebssysteme, verwendete Netzwerktechnologien und die Integrationsplattform erläutert.

Aufbauend auf der Analyse der Anforderungen und den identifizierten Anwendungsfällen wird in Kapitel 5 die Systemarchitektur erarbeitet, die die Grundlage für alle Entwicklungen darstellt.

Kapitel 6 befasst sich mit der Implementierung der einzelnen Module des Unterstützungssystems sowie der intranetbasierten Integrationsplattform. Die Implementierung erfolgt in iterativen Entwicklungsphasen, in denen durch Tests in der Fertigungsumgebung die Systemergonomie sichergestellt wird.

Die Erfahrungen mit dem Einsatz des Unterstützungssystems sowie eine Validierung unter verschiedenen Aspekten werden in Kapitel 7 beschrieben.

Kapitel 8 fasst die Arbeit zusammen und gibt einen Ausblick auf künftige IT-Systeme in der industriellen Fertigung.

2 Fertigungsorganisation im Schiffbau

Die Herstellung eines Schiffes erfordert heutzutage die nahtlose Synthese einer großen Anzahl von Fertigungsschritten. Das Produkt, in diesem Fall Containerschiffe mit einem Fassungsvermögen von bis zu 6.600 Standardcontainern, besteht aus ca. 500.000 Einzelkomponenten, die insgesamt sieben Montagephasen in der Produkthierarchie als Teilprodukte durchlaufen, bevor sie schließlich zum Endprodukt verschmelzen. Die Fertigungszeit eines solchen Schiffs beträgt ca. drei Monate.

Die Komplexität des Endproduktes – ein Schiff ist zu großen Teilen durch mehrfach gekrümmte Außenflächen begrenzt und besteht aus einer Vielzahl sehr unterschiedlicher Module – bewirkt eine sehr hohe Varianz der Teilprodukte. Zwar werden meist mehrere gleiche oder ähnliche Schiffe in Folge gebaut, jedoch ist eine Zwischenlagerung von Teilprodukten aufgrund der erheblichen Ausmaße und des begrenzten Lagervolumens nicht möglich. Werften müssen aus diesen Gründen weitestgehend auf Einzelteifertigung ausgelegt sein. Eine Einzelteifertigung dieser Größenordnung stellt besondere Anforderungen an Konstruktion, Planung sowie an die Flexibilität der Fertigung.

2.1 Traditionelle Organisationsform im Schiffbau

2.1.1 Auftragsmanagement und -bearbeitung

Die zeitliche und materialwirtschaftliche Planung der Fertigungsschritte basiert in der hier betrachteten Werft auf einem klassischen Top-Down-Ansatz. Das Produktmodell wird nach der Konstruktionsphase schrittweise in immer kleinere Einheiten zerlegt, bis der erforderliche Granulierungsgrad erreicht ist. Die Teilaufträge der untersten Ebene werden dann einzelnen Werkstätten und den zugehörigen ERP-Datensätzen (Enterprise Resource Planning) zugeordnet, sodass bereits in dieser Planungsphase der gesamte zeitliche und räumliche Verlauf der Produktion festgelegt wird. Alle Teilaufträge werden nun mit Terminen versehen an die Werkstätten vergeben.

Die Verteilung der Werkstattaufträge und somit auch die Aufteilung von Ressourcen sowie mögliche Kommunikationskanäle wird in Abbildung 2-1 aufgezeigt.

In der Planungsebene A werden Werkstattaufträge aus Produkt- oder Teilproduktaufträgen generiert. Bei komplexen Produkten, wie zum Beispiel im Schiffbau, entsteht hier aus dem Produktdatenmodell (PDM) durch mehrere Iterationen ein Baum vom Teilmodellen. Die daraus erzeugten Werkstattaufträge werden zentral dem Produktionsleiter der jeweiligen Werkstatt übertragen. Der Produktionsleiter sichtet die Aufträge und verteilt sie entsprechend der enthaltenen Prozesse und freier Ressourcen an die Vorarbeiter (Primäre Auftragsvergabe). Die Auftragsvergabe wird vom Produktionsassistenten unterstützt, der Kosten überwacht und Konflikte bei der Aufgabenverteilung zu lösen versucht.

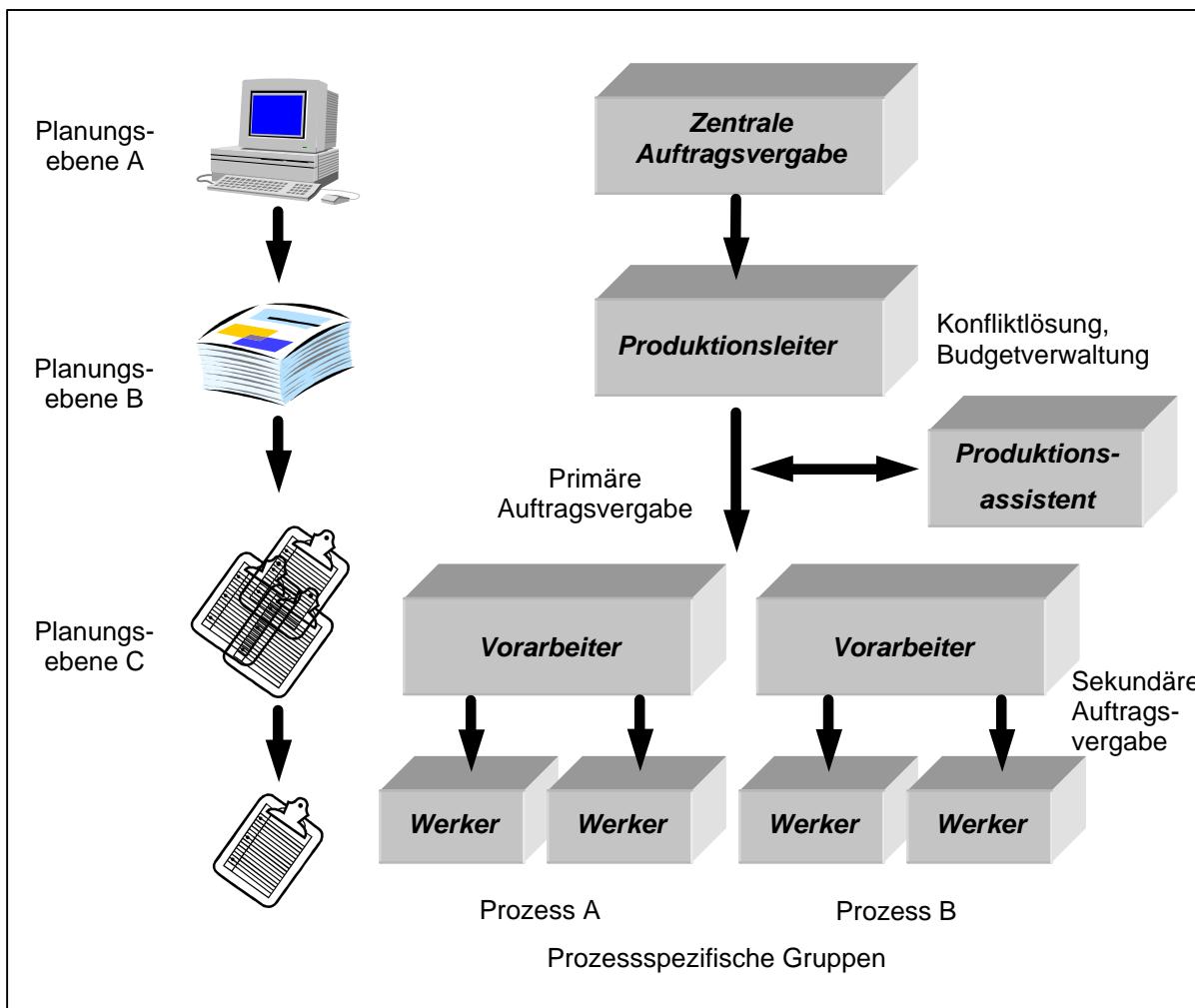


Abbildung 2-1: Auftragsvergabe in hierarchisch organisierter Werkstatt

Die Vorarbeiter vergeben nun Teilaufträge an die einzelnen Werker. Dieser Vorgang wird als „Sekundäre Auftragsvergabe“ bezeichnet. Das Werkstück ist nach

Beendigung einer Teilaufgabe nicht fertiggestellt, sondern muss zu einem weiteren Bearbeitungsschritt über den Vorarbeiter einem anderen Prozess zugeführt werden.

Ein solches Organisationsprinzip besitzt systemimmanente Merkmale, die großen Einfluss auf die Fertigungseffizienz haben. Die wichtigsten Merkmale sind:

- **Feste Zuordnung der Werker zu Prozessen**

Durch die feste Zuordnung der Werker zu spezifischen Prozessen an immer gleichen oder sehr ähnlichen Werkstücken entsteht hier eine monotone, wenig herausfordernde Tätigkeit. Die Folgen sind geringe Motivation und mangelndes Verständnis für Nachbarprozesse, sodass oft für andere Prozesse wichtige Kenntnisse über Eigenheiten eines Werkstücks verloren gehen.

- **Feste Zuordnung der Werker zu Fachgruppen**

Die feste Zuordnung identisch qualifizierter Werker zu Gruppen verhindert eine optimale Auslastung der Ressourcen, da zwischen den Gruppen kaum weitere Koordination nach Vergabe der Aufträge stattfindet. Ausfälle von Ressourcen können nicht ausgeglichen werden und führen oft zu Verzögerungen im Fertigungsablauf. Die Werker haben aufgrund der identischen Ausbildungen kaum Möglichkeiten der Weiterbildung in anderen Gebieten und Synergieeffekte zwischen Werkern unterschiedlicher Ausbildung werden verhindert.

Ein Vorteil der festen Zuordnung der Werker zu bestimmten Tätigkeiten und Fachgruppen ist die damit einhergehende Konzentration von prozess- oder fachspezifischem Wissen. Die Ausbildung von Experten wird somit gefördert.

- **Hierarchische Struktur der Auftragsvergabe**

Die hierarchische Struktur der Auftragsvergabe verringert die Flexibilität. Das Vermögen, auf kurzfristige Änderungen der Ressourcen- oder Auftragssituation reagieren zu können ist eingeschränkt, da auf Werkerebene kaum Entscheidungen getroffen werden können. Selbst geringfügige Planabweichungen müssen mit dem Vorarbeiter besprochen werden. Die erhebliche Ausdehnung der Produktionsstätten im Schiffbau mit teilweise mehreren hundert Metern Hallenlänge und der dort herrschende Lärm erschweren dem Vorarbeiter die Übersicht. Zudem ist er lediglich über den statischen Produktionsplan informiert und besitzt keine Möglichkeit, zentral den aktuellen Status der Aufträge einzusehen.

Ein positiver Aspekt der hierarchischen Struktur der Auftragsvergabe ist die klare Rollenzuweisung des Personals. Eindeutig definierte Positionen und Rangfolgen legen Autoritätsstufen fest und vereinfachen das Verständnis des Produktionsablaufs. Es herrscht ein hohes Maß an Solidarität und Kollegialität unter in der Hierarchie gleichgestellten Werkern.

- Hierarchische Struktur der logistischen Organisation

Unter logistischer Organisation wird die Projektion des Teilproduktflusses auf die zeitliche Struktur des Fertigungsprozesses verstanden. Traditionell herrscht auch hier eine hierarchische Organisationsform vor, sodass verschiedene Informationsdefizite beim Werker entstehen. So werden zum Beispiel Termine für die Fertigstellung eines Produktes nicht bis in die unteren Hierarchieebenen weitergegeben, sondern in Termine für Teilprodukte aufgeteilt. Treten unerwartet Probleme auf, kann der Fertigungsablauf nicht auf niedriger Ebene an die neue Situation angepasst werden, da die Werker weder über die Abhängigkeiten der Teilprodukte untereinander noch über den Fertigstellungstermin des Gesamtproduktes informiert sind.

Als Folge der fehlenden Koordination auf Werkerebene müssen auf Planungsebene die zu kalkulierenden Zeitintervalle der einzelnen Fertigungsschritte erhöht werden, um eine akzeptable Planungssicherheit auch im Falle von Problemen zu gewährleisten. Die Fertigungsduer für ein Produkt verlängert sich somit, da der nächste Fertigungsschritt erst für einen Zeitpunkt nach der zu großzügig geplanten Fertigstellung der Teilprodukte geplant wurde. Üblicherweise werden Aufträge nicht vorgezogen, sondern die Aufgaben bearbeitet, die im nach Datum sortierten Produktionsplan anstehen.

Aus den verlängerten Zeitintervallen der einzelnen Fertigungsschritte ergeben sich Totzeiten. Dies bedeutet, dass das Teilprodukt nicht direkt der nächsten Arbeitsstation zugeführt werden kann. Die Folge ist insbesondere bei der Übergabe eines Teilproduktes zur Folgewerkstatt die Ausbildung von unstrukturierten Zwischenlagern als Puffer zwischen den verschiedenen Fertigungsschritten. Der Transport der Teilprodukte und die Suche nach dem richtigen Teilprodukt im Zwischenlager kostet zusätzlich Zeit und Ressourcen.

Der Einfluss der Totzeiten summiert sich besonders im Schiffbau durch die Komplexität des Produktes und durch die zahlreichen Abhängigkeiten innerhalb des Teilproduktbaums zu beträchtlichen, bereits im Voraus eingeplanten Verzögerungen auf.

Zudem gehen während der organisatorisch bedingten Totzeit zwischen einzelnen Fertigungsschritten Informationen der Werker zum Beispiel über Besonderheiten des Werkstücks oder Fertigungsprobleme verloren, da aufgrund der zeitlichen Differenz keine Übergabe des Werkstücks stattfindet und ein die Totzeit überbrückendes Kommunikations- bzw. Dokumentationssystem fehlt. Nur selten werden solche Informationen an die Planungs- bzw. Konstruktionsabteilung oder an den nachfolgenden Prozess weitergegeben. Es kommt zu Wiederholungen

derselben Fehler. Der Fertigungsprozess wird nicht verbessert und gleichzeitig führt diese Situation zu einer Resignation der Werker.

Neben diesen negativen Argumenten muss gleichzeitig darauf hingewiesen werden, dass im Schiffbau die Planungssicherheit eine hohe Priorität einnimmt. Kommt es zu Verzögerungen der geplanten Auslieferung eines Schiffs, sind hohe Konventionalstrafen branchenüblich. Die zeitlich großzügige Planung ist daher beim Einsatz eines solchen Planungssystems notwendig.

Der beschriebene Top-Down-Ansatz ist zwar theoretisch umsetzbar, jedoch kann durch die Nichtbeachtung stochastischer Elemente wie zum Beispiel Ungenauigkeiten, Ausfall von Ressourcen und unterschiedlicher Materialeigenschaften eine optimale Auslastung der vorhandenen Fertigungskapazitäten nicht gewährleistet werden. Die Verzögerungen durch oben angeführte Einflüsse bewegen sich in der Größenordnung der ursprünglich geplanten Dauer des Teilprozesses.

Das streng hierarchisch strukturierte Auftragsmanagement hat über Jahre die organisatorische Auslegung der Werft geprägt, sodass selbstverantwortliches Handeln, umfassendes Verständnis der Aufgabe sowie zielgerichtete Motivation auf Werkerebene nur schwach ausgeprägt sind [VML98]. Es findet nur wenig Kommunikation unter Werkern sowie zwischen Werkern und z.B. der Planungs- oder Konstruktionsabteilung statt. Wichtige Informationen über Probleme bei der Fertigung, Verbesserungsvorschläge bezüglich Konstruktion und Planung werden oft nicht genutzt.

Die Kombination aus hierarchischer Planung ohne genügende Kenntnis des aktuellen Zustandes der Fertigungsprozesse und organisatorischer Struktur ohne notwendige Entscheidungskompetenzen, auf Störungen zu reagieren, zwingt die Werft, Prozesszeiten einzuplanen, die weit über den tatsächlich benötigten Zeiten liegen. Durch die zu lang ausgelegten Prozesszeiten erhöht sich zwar die Planungssicherheit, jedoch wird die Einrichtung von Zwischenlagern für Teilprodukte erzwungen und Wartezeiten erhöht, sodass sich die Fertigungsdauer des Endproduktes signifikant verlängert. Die Verlängerung der Gesamtproduktionsdauer wirkt sich direkt auf Produktionskosten und Reaktionszeit der Werft („Time-to-Market“) aus. Im stark global orientierten Schiffbau sind dies wichtige Faktoren, um auf dem Markt vor allem gegen die Konkurrenz aus Asien bestehen zu können.

In den folgenden Kapiteln werden die zwei Werkstätten der Werft beschrieben, an deren Beispiel die derzeitige Situation analysiert wird. Basierend auf dieser Analyse wird später ein Konzept zur Effizienzsteigerung auf Fertigungsebene hergeleitet und umgesetzt. Durch die Wahl zweier unterschiedlicher Werkstätten soll sichergestellt werden, dass das in dieser Arbeit entwickelte Konzept und seine Umsetzung nicht an

einen bestimmten Werkstatt-Typus gebunden ist, sondern vielmehr ein generisches System darstellt, das auf andere Fertigungsstrukturen übertragbar ist.

2.1.2 Beispiel Rohrwerkstatt

Für ein durchschnittliches Containerschiff müssen ca. 11.000 Rohre angefertigt werden, die jeweils aus Rohrstücken, Flanschen und Reduzierstücken bestehen (Abbildung 2-2). Rohre können mehrfach in verschiedenen Ebenen gekrümmt sein und Verzweigungen beinhalten. Die Durchmesser reichen dabei von zwei bis zu 24

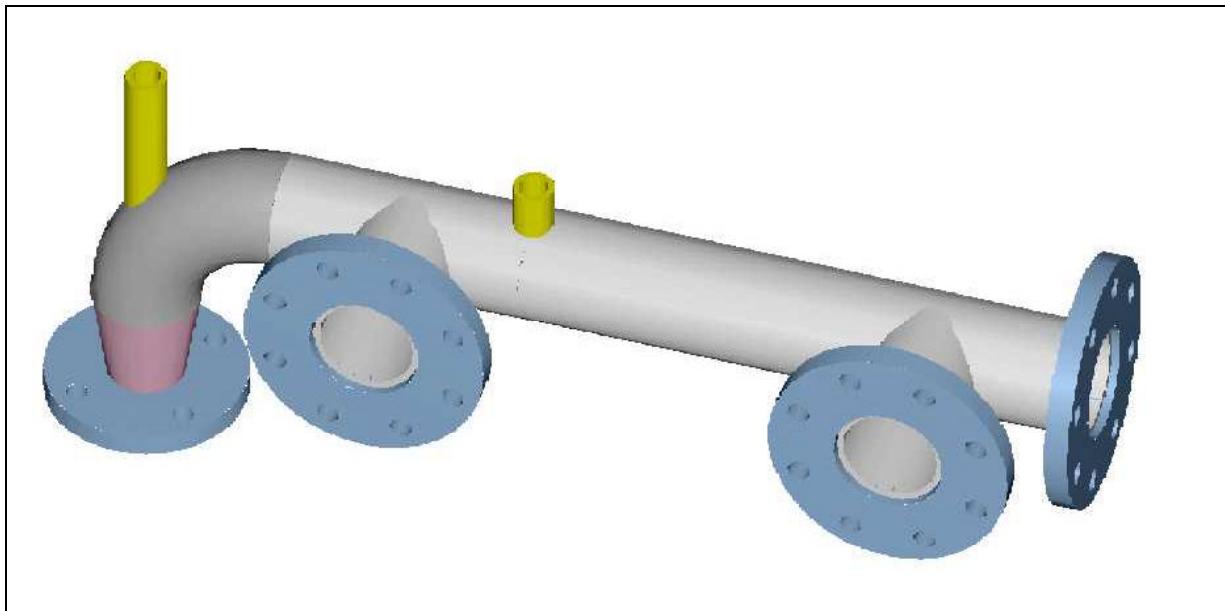


Abbildung 2-2: Typisches Beispiel eines Frischwasserrohrs mit Verzweigungen, Reduzierstück und Flanschen

Zoll und können an einem Bauteil in Kombination auftreten. Die Rohre werden im wesentlichen für Trink-, Kühl- und Heizwasser, für verschiedene Öle sowie für Aufbauten wie z.B. Reling und andere Geländer verwendet.

Die Fertigung von Rohren gilt als einer der komplexesten Prozesse im Schiffbau. Insbesondere das Biegen, das Schneiden von Löchern und das Zusammenschweißen der Segmente ist oft mit Schwierigkeiten verbunden, da hier hohe Anforderungen an das räumliche Vorstellungsvermögen der Werker gestellt werden.

2.1.2.1 Rohrkonstruktion

Die große Produktvielfalt erfordert hier ein sehr detailgetreues Produktdatenmodell

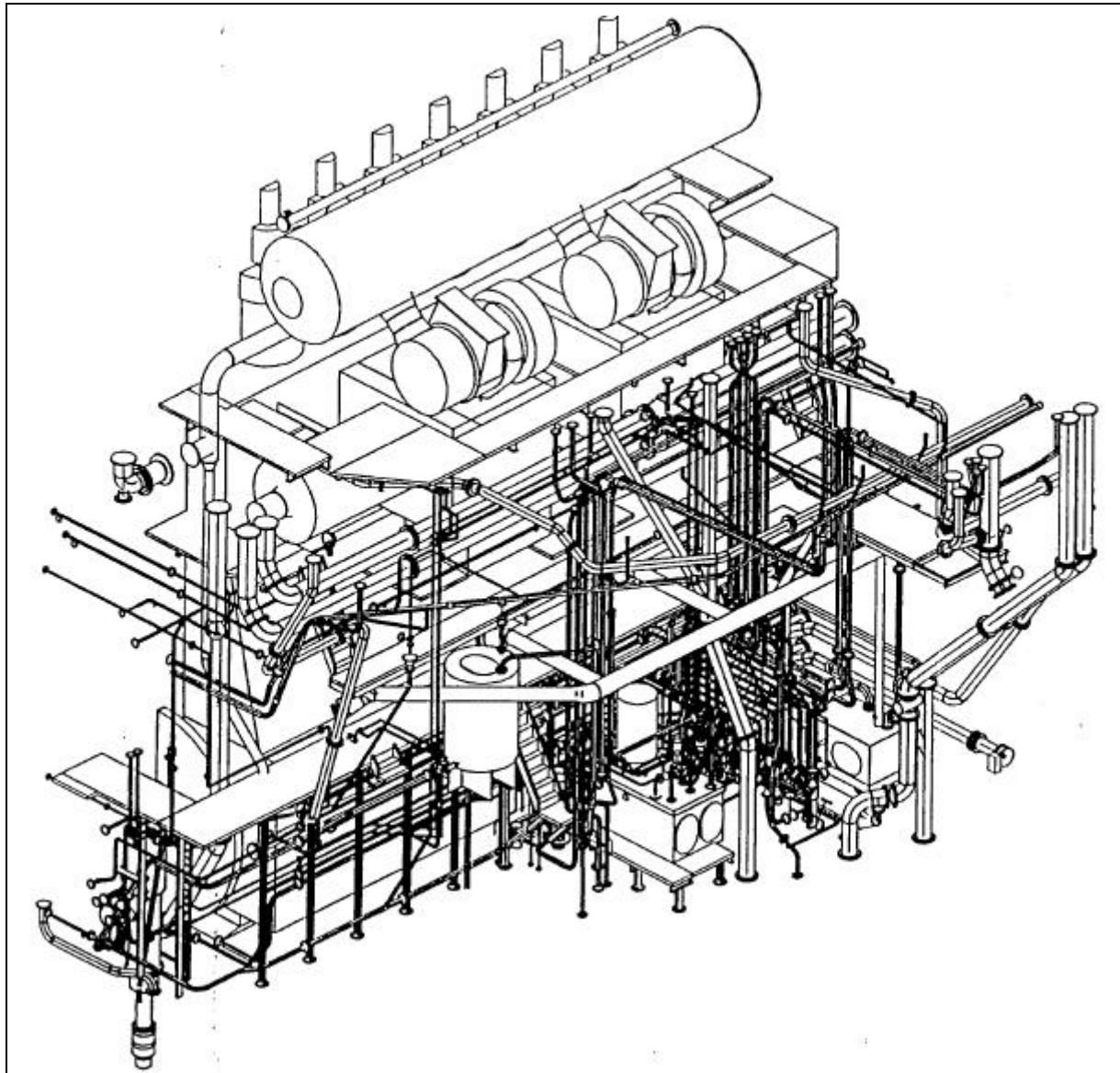


Abbildung 2-3: CAD-Modell der Rohre eines Schiffsegments

und ein effizientes Verfahren zur Handhabung der Daten vom Entwurf bis hin zur Produktion. Das Produktdatenmodell basiert auf dem in der Werft zum Einsatz kommenden CAD-System Hicadec-P, das Rohrverlauf, Durchmesser und Position der Rohre im Schiffsrumph definiert. Ein Beispiel eines Teils des Produktdatenmodells ist in Abbildung 2-3 dargestellt.

Das verwendete CAD-System stellt den optimalen Verlauf der Rohre innerhalb des Schiffes sicher und führt anschließend eine Kollisionsüberprüfung durch, um Überschneidungen von Rohren zu vermeiden. Das CAD-System verfügt über eine direkte Verbindung zum Materialbeschaffungssystem. Zuletzt wird der Zusammenbau der einzelnen Rohre geplant. Da kein Produktzustandsmodell verwendet wird, ist ein Vergleich zwischen geplanter und aktueller Produktion nur schwer möglich. Es kommt zu sich immer wiederholenden Fehlern gleichermaßen auf Planungs- und Produktionsebene. Beispielsweise werden Verzögerungen durch konstruktionsbedingte Fertigungsprobleme nicht der Konstruktions- oder Planungsabteilung gemeldet.

Das eingesetzte System PMS (Production Management System) beschränkt sich auf die Planung der Zusammenhänge zwischen vorgefertigten Rohren. Die Fertigung der Rohre selbst wird nicht von diesem System erfasst. Der zeitliche Ablauf des Zusammenbaus der Rohre wird erst dann festgelegt, wenn alle betreffenden Rohre des Schiffsegments fertiggestellt sind. Prozessbedingte Abweichungen vom Plan können somit nicht in das PMS aufgenommen werden und es kommt zu weiteren Störungen des Produktionsplans.



Abbildung 2-4: Rohrpaket für ein Schiffssegment

2.1.2.2 Werkstattplanung

Das PMS erzeugt aus den Geometriedaten einen groben Produktionsplan und die Materialliste, die die Grundlage für die Materialanforderung darstellt. Der Produktionsplan wird der Rohrwerkstatt mit einem Planungshorizont von zwei Wochen übergeben. Die Planung der Rohrproduktion ist in Abbildung 2-5 dargestellt. Innerhalb der Rohrwerkstatt erfolgt nun die Auswahl der zu verwendenden Ressourcen und die Zeitplanung für jede einzelne Station. Basierend auf dem Zeitplan wird der Zuschnitt des angelieferten Materials berechnet und eine Liste aller innerhalb einer Woche zu schneidenden Rohre ausgedruckt.

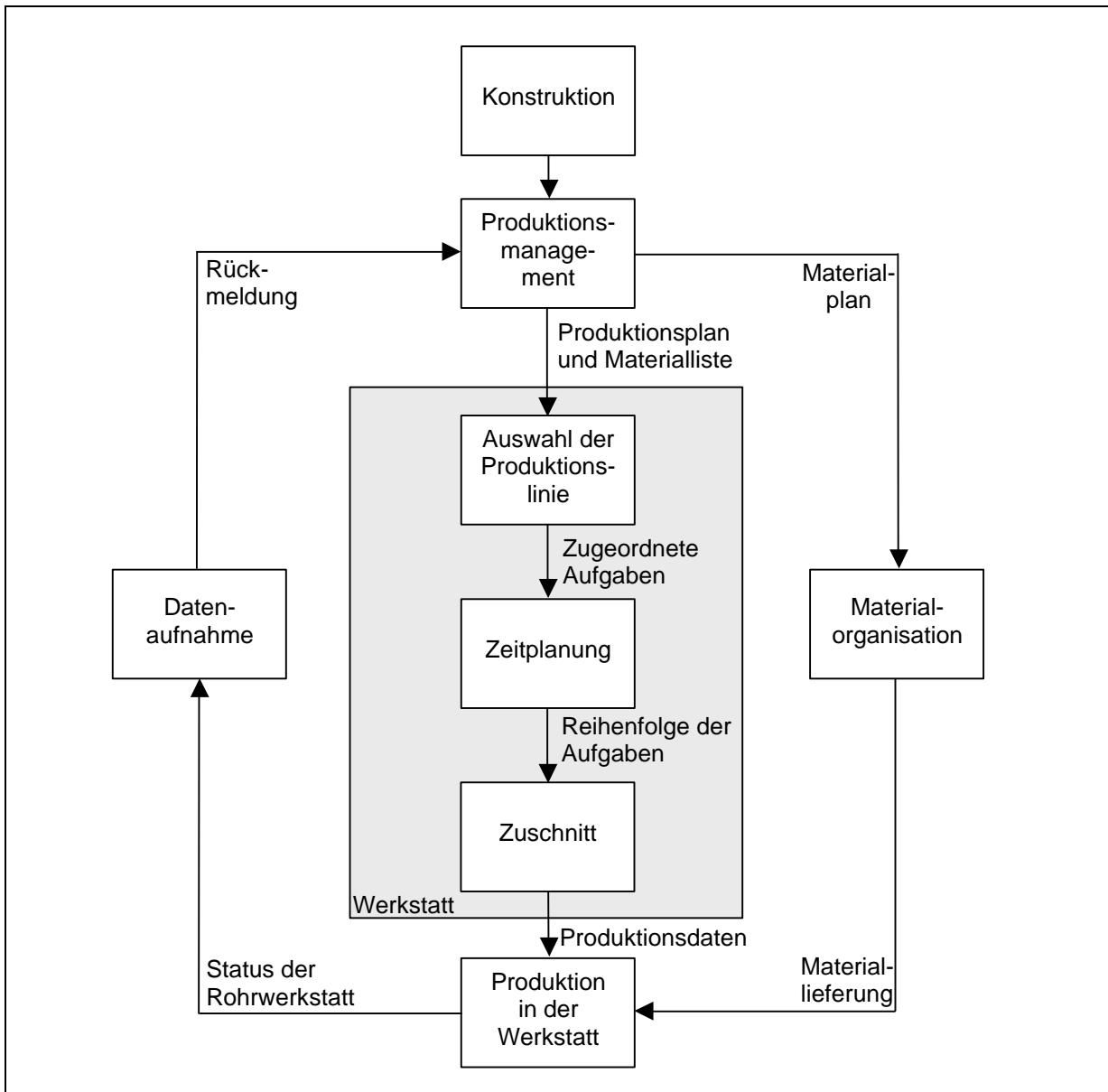


Abbildung 2-5: Planung der Rohrproduktion

Für die Fertigung eines Rohres werden aus dem CAD-System die in Abbildung 2-6 gezeigten 2D-Skizzen erzeugt. Diese Skizzen sollen dem Werker vor Ort als Hilfe bei der räumlichen Vorstellung des fertigen Endproduktes dienen. Sie sind jedoch so stark vereinfacht, dass oftmals weitere Hilfsmittel notwendig sind, um sich ein Bild der zu fertigenden Rohre zu machen. Meist fertigen die Werker vor Beginn der Produktion kleine Rohrmodelle aus Drahtstücken an, anhand derer sie die Stellung der Rohrabschnitte zueinander erkennen können.

Neben den Skizzen werden vom Planungssystem maschinenspezifische Daten generiert, die in Datenbanken hinterlegt direkt den jeweiligen Maschinen zur Verfügung stehen. Diese Daten bestehen im wesentlichen aus dem für das einzelne

Rohr erzeugten NC-Programm für die Schneidemaschine, die Biegemaschine und die Flansch-Schweißmaschine.

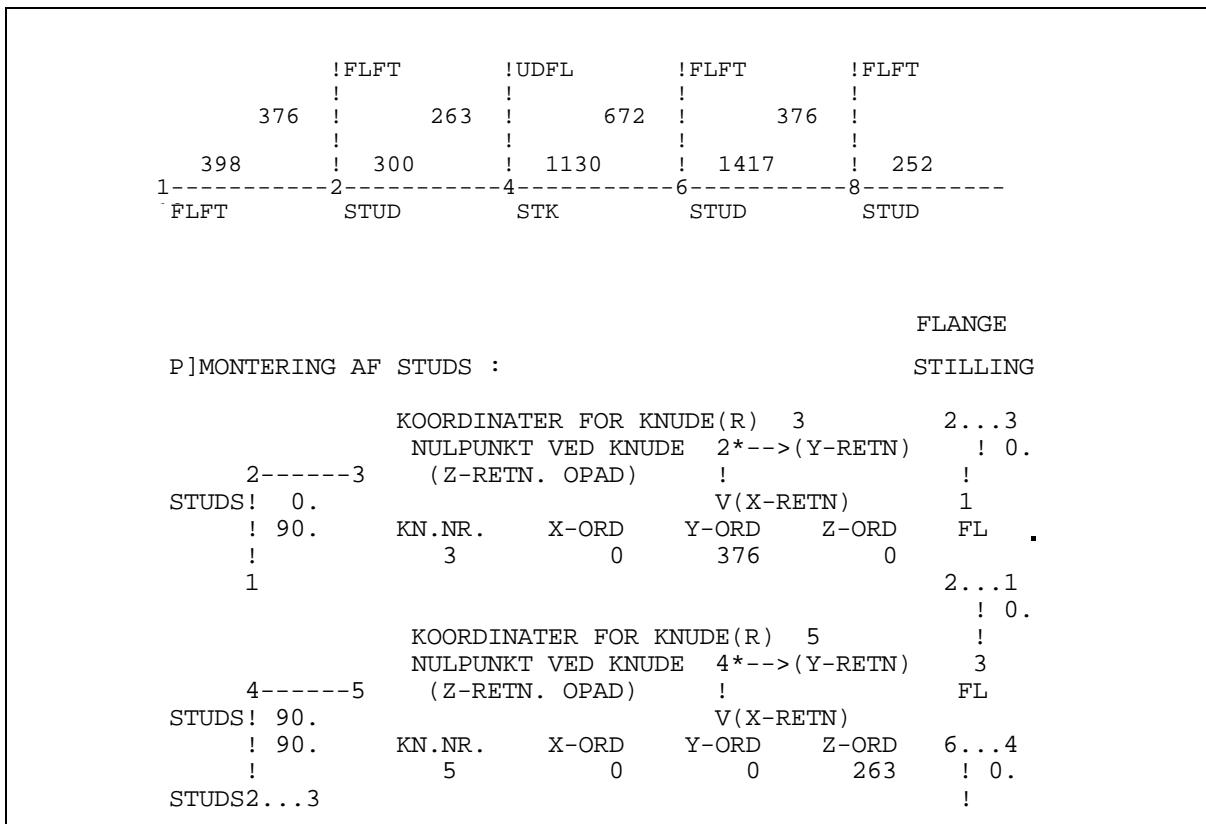


Abbildung 2-6: Ausgabe des Planungssystems – „2D-Skizze“

Der Werker ist auf die in den in Abbildung 2-6 dargestellten Skizzen enthaltenen Informationen angewiesen. Auf Basis der skizzenhaft vorliegenden Geometriedaten wird üblicherweise mit Hilfe von Pfeifenreinigern aus Draht ein kleines dreidimensionales Modell des zu fertigenden Rohres hergestellt, um Winkel und relative Stellung der Rohrabzweigungen zueinander erkennen zu können. Es existiert heute für den Werker keine weitere Möglichkeit, sich beispielsweise das Zielprodukt anzeigen zu lassen oder sich über den aktuellen Zustand eines Auftrags zu informieren. Ein zusätzlicher Nachteil ist das Fehlen eines Kommunikationssystems, das z.B. das Übermitteln von Fehlerberichten oder Verbesserungsvorschlägen erlauben würde.

2.1.2.3 Rohrfertigung

Die Abfolge der Fertigungsprozesse ist in Abbildung 2-7 vereinfacht dargestellt. Aus dem Rohrlager wird entsprechend der PMS-Daten Rohmaterial entnommen. In der Vorbereitungsphase wird zwischen Rohren mit einem Durchmesser von über acht und unter acht Zoll unterschieden. Rohre über acht Zoll werden an einer Plamaschneideanlage geschnitten und direkt der manuellen Vormontagestation

zugeführt. Krümmungen dieser Rohre werden nicht durch Biegen erzeugt, sondern mithilfe von schräg geschnittenen Rohrsegmenten, die an der manuellen Montagestation verschweißt werden. Dort werden sie auch mit Flanschen und Reduzierstücken versehen.

Rohre unter acht Zoll werden zunächst gesägt und an den Schnittkanten geschliffen. Mithilfe der folgenden automatischen Schweißmaschine werden sie mit Flanschen versehen und dann an einer CNC-Biegemaschine gebogen. Die fertiggestellten Rohrabschnitte werden an der manuellen Montagestation mit weiteren eventuell vorgefertigten Teilen oder Rohren zusammengefügt.

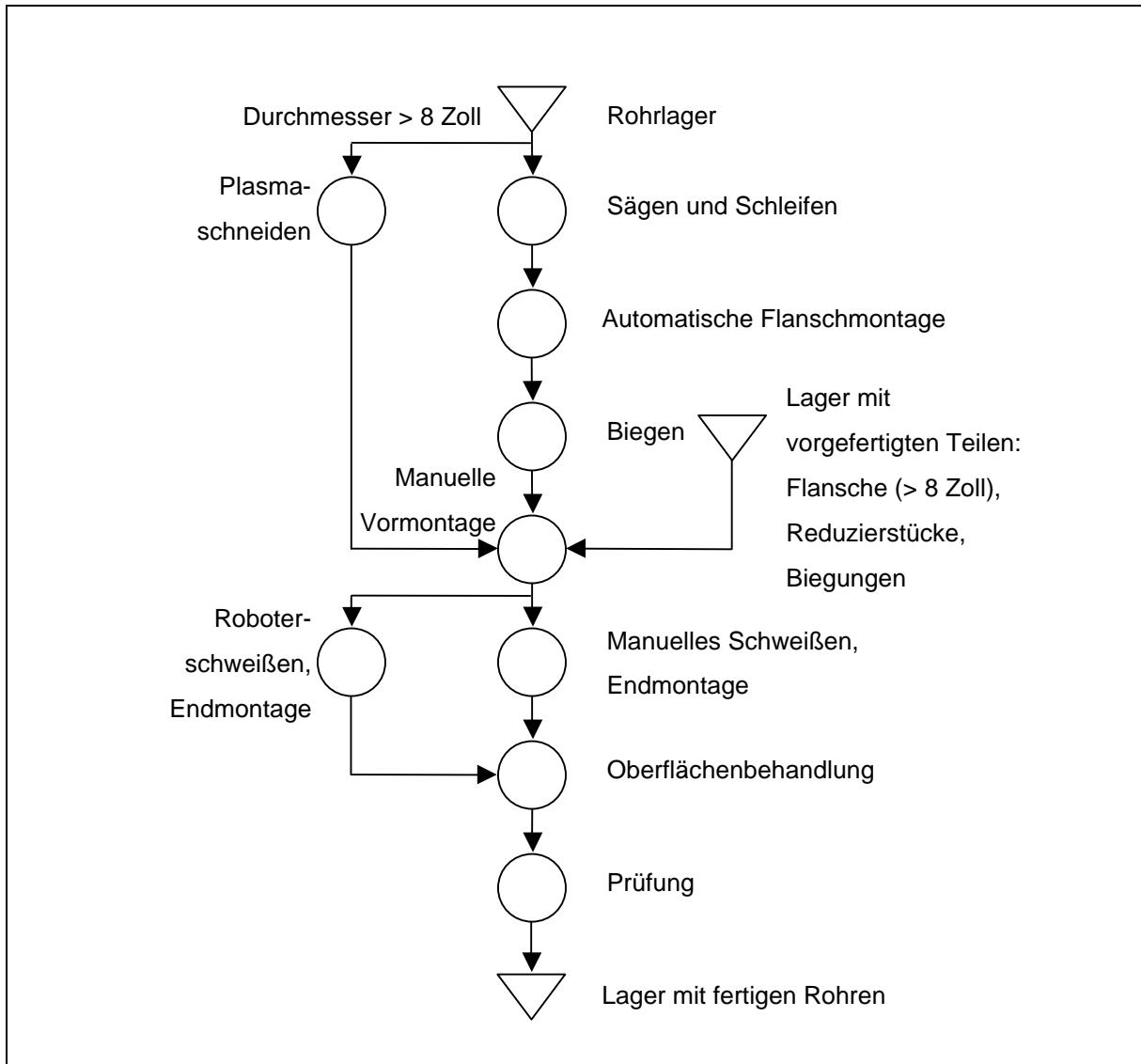


Abbildung 2-7: Prozesskette der Rohrwerkstatt

Die vorgefertigten Rohrsegmente werden nun von der manuellen Vormontage zur manuellen oder automatisierten Endmontage transportiert. Anschließend erfolgt in Abhängigkeit vom späteren Verwendungszweck eine Oberflächenbehandlung, die

Reinigen, Grundieren und Lackieren umfassen kann. Bevor die fertigen Rohre in die folgende Werkstatt transportiert werden, werden sie geprüft und in einem Zwischenlager aufbewahrt.

2.1.3 Beispiel Montagewerkstatt für Strukturelemente

Als zweites Beispiel für eine industrielle Werkstatt soll hier eine Montagewerkstatt für Schiff-Strukturelemente untersucht werden. Diese Werkstatt wurde aufgrund der fertigungstechnischen und organisatorischen Unterschiede zur bereits beschriebenen Rohrwerkstatt ausgewählt.

Die für den Schiffsrumpf und die Aufbauten benötigten Elemente bestehen ausschließlich aus Stahlplatten, die hier zu Profilen und Segmenten zusammengeschweißt werden.

Das Gewicht der Produkte dieser Werkstatt liegt zwischen 1 und 15 Tonnen bei einer Größe von $2 \times 2 \times 1$ bis $16 \times 3 \times 4,5$ Metern. Ein solches Element besteht aus 2 bis 30 Teilen. Größe, Gewicht und Komplexität der Werkstücke stellen besondere Anforderungen an den Fertigungsablauf. Räumliche Randbedingungen müssen in die Werkstattplanung mit einbezogen werden, da die Anzahl der Werkstücke, die einer Arbeitsstation gleichzeitig zugeführt werden können von ihrer jeweiligen Ausdehnung abhängt. Abbildung 2-8 zeigt ein typisches Beispiel für ein solches Strukturelement.

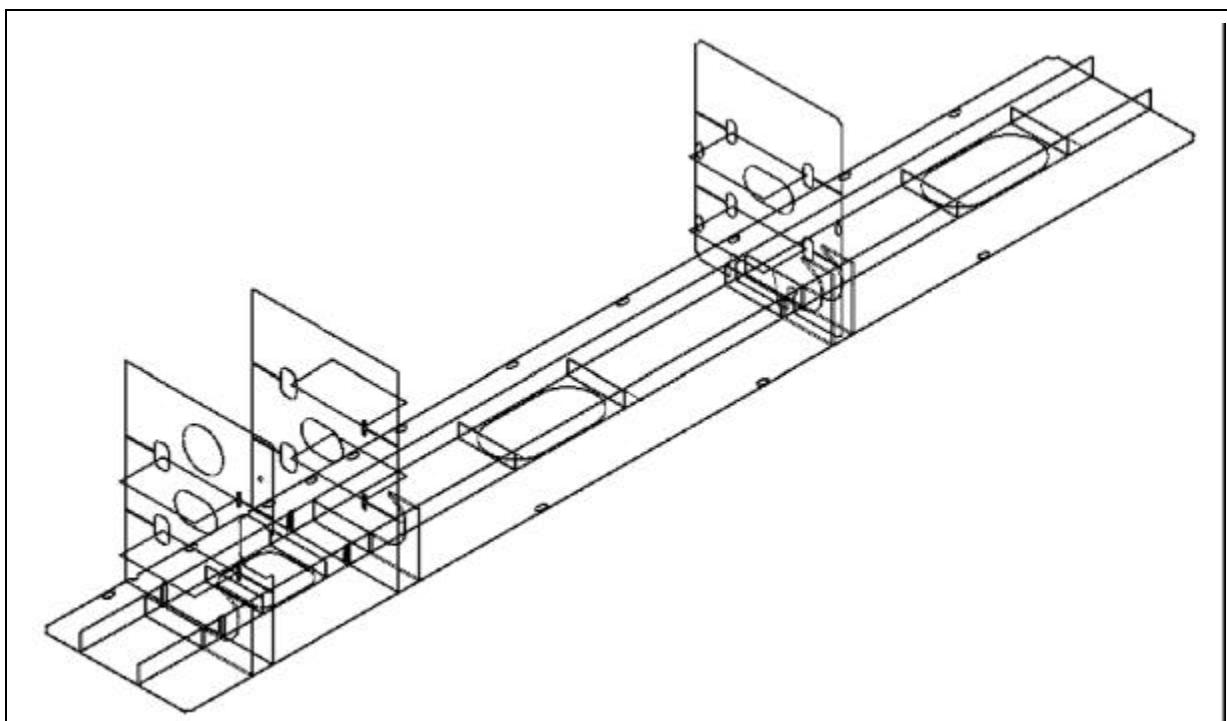


Abbildung 2-8: Beispiel für ein Schiffs-Strukturelement

Die Elemente werden in einem oder zwei Schritten gefertigt. Das oben gezeigte Element wurde in zwei Schritten zusammengesetzt: Die Basisplatte sowie die drei großen Querplatten wurden in einem ersten Schritt vorgefertigt, um im zweiten Schritt zum Endprodukt montiert zu werden.

2.1.3.1 Fertigungsplanung

Die in dieser Werkstatt eingesetzten automatischen Schweißmaschinen und Roboter werden mit Daten aus der Konstruktionsabteilung gespeist und können so unter Einsatz integrierter Sensorik selbständig Teile des Fertigungsprozesses ausführen.

Die Planung auf Werkstattebene sowie die Koordination der automatisierten Prozesse unterliegt jedoch der Werkstattleitung. Sie entscheidet auf Basis der aktuellen Auftragslage, der angeforderten Produkte und der Verfügbarkeit von Ressourcen über den Zeitpunkt der Ausführung von Aufträgen innerhalb der Werkstatt.

Die unterste Planungsebene der Werft wird als C-Plan bezeichnet. Im C-Plan werden alle von einer Werkstatt zur bearbeitenden Aufträge mit Terminen für ihre

C-PROGRAM NR. : 15			GROUP : 15		AFD. : 1104-5J		UDSENDT : 19970625					
FORDELING : VGN, SKM, BDS, SK, JSP, JNO, HL			EMNER : DELELEMENTER		SIDE : 1 AF 3		REF. PLAN : 97-03-00					
			HJNEO		AUG		SEP SEP					
NR.	SG	SEKT	Pj	TEKST	PSK	14	08	16	22	28	34	PLA
001	162	2764B	G	TR (LBH)	1	14320	1	14410	1	14420	1	10.9 11.8 5.0
002	162	3764B	F	TR + 3D (LBH)	2	14320	1	14410	1	14420	1	10.9 11.8 5.0
003	162	2156A	C	G+TR (DB)	3	14320	1	14410	1	14420	1	149.2 162.0 4.0
004	162	3156A	B	G+TR (DB)	0	14320	1	14410	1	14420	1	186.6 203.6 34.6
005	162	22640	F	TR (5d)	1	14320	1	14410	1	14420	1	1.8 2.0
006	162	2764F	D	LBH	2	14320	1	14410	1	14420	1	6.4 8.6
007	161	2270A	M	LBH	1	14320	1	14410	1	14420	1	8.2 6.4
008	161	3270A	G	TR	2	14320	1	14410	1	14420	1	8.2 6.4
009	161	1350A	B	LBH+40+60	3	14320	1	14410	1	14420	1	62.8 51.0 5.0
010	162	2265A	A	TR+4+6+70 (LBH)	1	14320	1	14410	1	14420	1	51.0 54.6
011	162	3764A	C	LBH	1	14320	1	14410	1	14420	1	2.7 3.6
012	162	3265A	A	TR+4+6+70 (LBH)	2	14320	1	14410	1	14420	1	51.0 54.6
013	161	1361	B	TR+LBH	1	14320	1	14410	1	14420	1	60.1 41.9 5.0
014	161	1370A	A	TR+LBH+4+6D (TV, SK, 1)	2	14320	1	14410	1	14420	1	38.2 30.0 5.0
015	162	3764G	E	LBH	1	14320	1	14410	1	14420	1	8.2 8.9
016	162	2266A	A	TR+4+60 (LBH)	2	14320	1	14410	1	14420	1	53.7 57.3
017	162	3266A	A	TR+4+60 (LBH)	1	14320	1	14410	1	14420	1	53.7 57.3
018	162	2147A	B	G+TR (DB)	2	14320	1	14410	1	14420	1	169.3 153.8 45.0
019	162	3147A	B	G+TR (DB)	1	14320	1	14410	1	14420	1	213.5 194.7 345.0
020	162	1345	B	TR	1	14320	1	14410	1	14420	1	145.6 119.2 5.0
021	162	2764C	A	LBH (2D)	1	14320	1	14410	1	14420	1	0.9 0.9
022	162	3764F	J	TR+LBH (UD)	1	14320	1	14410	1	14420	1	4.6 4.6
023	162	1346	B	TR	1	14320	1	14410	1	14420	1	155.6 131.9
024	162	3764C	H	TR	2	14320	1	14410	1	14420	1	3.7 3.6

Abbildung 2-9: Beispiel eines C-Plans

Fertigstellung aufgelistet. Die Organisation der Werkstatt obliegt den Werkstattleitern und wird nicht durch rechnerbasierte Methoden unterstützt. Der C-Plan wird im DPS

(Decentralised Planning System) mit einem Horizont von zwei bis drei Wochen erzeugt und ist werksweit im Netzwerk verfügbar. Der Zugriff auf den Plan erfolgt mithilfe von UNIX-Terminals. Das bereits früher erwähnte PMS erzeugt Sortier- und Montagepläne, die den Werkstattleitern in Form von langen gedruckten Listen vorliegen. Abbildung 2-9 zeigt ein Beispiel für den Ausdruck eines C-Plans. Der Werkstattleiter hat hier mit Bleistift eine Änderung vermerkt, die jedoch nicht in das Rechnersystem eingetragen wird. Die Verbesserung zukünftiger Planungen sowie die Abstimmung mit der Folgewerkstatt werden dadurch erschwert.

Täglich werden zwischen 10 und 20 Produkte fertiggestellt. Die Bearbeitungszeit der Aufträge beträgt in der Regel vier bis fünf Tage. Die Verwendung eines speziellen Rollensystems, das für den Transport der Werkstücke durch die Werkstatt verwendet wird, erfordert eine fortlaufende Koordination der Aufträge, da das Material zu Beginn der Bearbeitung eines Auftrags relativ viel Fläche auf der Produktionsstrasse einnimmt, sein Flächenbedarf sich jedoch im Prozessverlauf mit der Montage der Teilprodukte stark verringert. Hinzu kommt, dass die Verweildauer der Teilprodukte an den unterschiedlichen Arbeitsstationen variiert. Der Werkstattleiter hat dafür Sorge zu tragen, dass einerseits das Rollensystem möglichst gut ausgelastet ist, gleichzeitig die Produkte aber die Werkstatt in der korrekten Reihenfolge verlassen, da die Kapazität des puffernden Lagers am Ende der Fertigungsstraße sehr begrenzt ist.

Zusammenfassend kann man feststellen, dass das Automatisierungspotenzial der Werkstatt nahezu ausgeschöpft wurde. Bei den nicht automatisierten manuellen Tätigkeiten wird der Werker nicht unterstützt. Er verfügt über keine Planungsinformationen, sondern führt seine Arbeit auf Anweisung der Vorarbeiter aus. Gleichzeitig ist die von der Werkstattleitung durchgeföhrte manuelle Planung sehr fehleranfällig und wird in der Regel nicht an veränderte Randbedingungen angepasst, da der Werkstattleitung weder aktuelle Prozessinformationen noch Informationen über mögliche Probleme übermittelt werden. Verbesserungsvorschläge der Werker oder eine Optimierung der Planung sind somit nicht ohne weiteres durchzuführen.

2.1.3.2 Fertigung

Die Werkstatt ist aufgrund des eingesetzten Prinzips der linearen Produktionsstrasse (Fließfertigung) erfolgreich in großen Bereichen automatisiert worden. Hauptsächlich Tätigkeiten an schwer zugänglichen Stellen und Spezialarbeiten werden noch manuell ausgeführt. Entlang der gemeinsamen Fertigungsstraße werden alle wesentlichen Fertigungsschritte ausgeführt. Sie ist in acht Stationen unterteilt, die in einer festgelegten Reihenfolge durchlaufen werden (Abbildung 2-10).

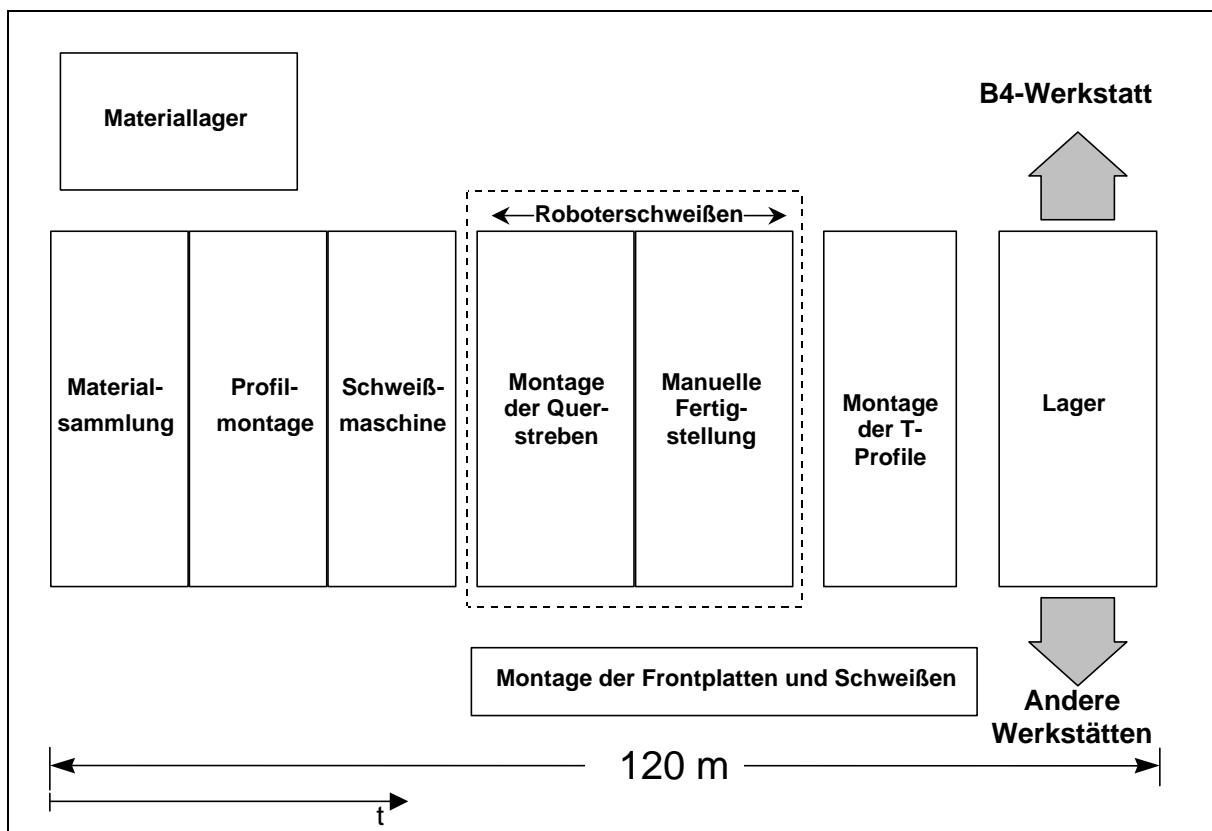


Abbildung 2-10: Fertigungsstationen der Montagewerkstatt

Die Werkstücke werden dabei mit dem automatischen Rollensystem zur folgenden Station transportiert und positioniert. Das Rollensystem ist in mehrere Segmente aufgeteilt, sodass die Werkstücke selektiv transportiert werden können. Die Steuerung des Transportsystems wird durch den Einsatz von laserbasierten Sensoren, die eine genaue Positionierung des Werkstücks ermöglichen, unterstützt.

Während der Materialsammlung wird das bereits zugeschnittene Material für ein bestimmtes Werkstück zusammengestellt und mit Hilfe eines Krans auf dem Rollensystem positioniert. Dabei werden zwar große Platten in der richtigen Reihenfolge angeliefert, kleine Platten müssen jedoch manuell herausgesucht und sortiert werden.

In der Station zur Profilmontage werden dann Haupt- und Untermodule zusammengesetzt und manuell durch Heftschweißungen fixiert. Von der darauffolgenden Schweißmaschine werden dann die langen Nähte der Profile automatisch geschweißt.

An der nächsten Station werden dann die Querstreben fixiert und entweder mithilfe von Robotern oder – bei erschwertem Zugang – manuell verschweißt. Die Roboter werden von großen Portalen mit drei Zusatzachsen geführt, sodass sich ein sehr großer Arbeitsbereich über die Fläche von zwei Arbeitsstationen ergibt.



Abbildung 2-11: Manuelle Profilmontage

Zusätzlich sind die Roboter mit magnetischen und optischen Nahtführungssensoren ausgerüstet, sodass sie den Verzug der Werkstücke im Bereich von einigen Zentimetern autonom ausgleichen können.

Anschließend wird das Werkstück aus den einzelnen Modulen zusammengesetzt und manuell oder mithilfe von Robotern fertiggestellt.

Besonders große T-Profile werden vorher entnommen und mit einem Kran zur entsprechenden Arbeitsstation transportiert, wo sie von Hand montiert werden (Abbildung 2-11).

Einige Werkstücke erfordern die Montage einer Frontplatte, die aufgrund der besonders schwierigen Erreichbarkeit manuell an einer neben der Fertigungsstraße

liegenden Arbeitsstation angebracht wird. Die fertigen Produkte werden anschließend zwischengelagert, bis sie von der nachfolgenden Werkstatt angefordert werden.

2.2 Soziale Aspekte und Kommunikation

Eine Grundlage für die Entwicklung eines Konzeptes für ein Produktionsunterstützungssystem ist die Analyse der zukünftigen Anwender. In beiden untersuchten Werkstätten wurden sehr ähnliche Workerprofile angetroffen, sodass hier die Ergebnisse teilweise zusammengefasst dargestellt werden können.

Es muss zwischen unterschiedlichen Rollen der Werker unterschieden werden. Die Rollen ergeben sich aus der Qualifikation, die der Werker im Laufe seiner Tätigkeit erlangt hat. Tabelle 2-1 zeigt eine Aufstellung der in den Werkstätten angetroffenen Qualifikationen.

Rohrwerkstatt		Montagewerkstatt	
Vorarbeiter	3	Vorarbeiter	3
Rohrschmiede	27	Schiffbauer	14
Schweißer	17	Schweißer	11
Allgemeines Personal	15	Allgemeines Personal	5
Gesamt	62	Gesamt	33

Tabelle 2-1: Werkerqualifikationen

Die Qualifikationen der Werker können in drei Kategorien zusammengefasst werden. Die Vorarbeiter sind im allgemeinen als Rohrschmied oder Schiffbauer ausgebildet und haben eine Weiterbildung an den Planungssystemen der Werkstätten erhalten.

Rohrschmiede und Schiffbauer haben einen vergleichbaren Qualifikationsgrad. Beide Berufe basieren auf einer vierjährigen Ausbildung in verschiedenen Werkstätten der Werft. Neben den umfangreichen fertigungstechnischen Kenntnissen sind sie meist mit dem Produktionsablauf vertraut und können komplexe Aufgaben bearbeiten. Die jüngeren Rohrschmiede und Schiffbauer besitzen meist Kenntnisse der in den Werkstätten eingesetzten Rechnersysteme und vereinzelt auch Erfahrungen mit Internettechnologie, während die älteren Werker zum Einsatz traditioneller, papierbasierter Systeme neigen.

In beiden Werkstätten werden Schweißer eingesetzt. Die Qualifikation der Schweißer besteht minimal aus einer einmalig abzulegenden Prüfung, im Regelfall jedoch aus einer vierjährigen Ausbildung, in der sie eine Vielzahl schweißtechnischer Fertigungsverfahren erlernen. Obwohl Schweißer eine vergleichbar lange Ausbildung erhalten wie Rohrschmiede oder Schiffbauer, bleibt ihr Einsatzbereich auf ihr Spezialgebiet beschränkt. Ihre Ausbildung umfasst nicht immer den Einsatz von Rechnern im Werkstattumfeld, oft trifft man aber auf Erfahrungen in diesem Bereich.

Als „allgemeines Personal“ werden Werker ohne spezielle Qualifikation oder mit fachfremder Ausbildung bezeichnet. Sie verrichten meist Hilfs- und Transportarbeiten.

In Bezug auf die Erfahrungen mit Rechnertechnik kann eine starke Abhängigkeit von Alter und Qualifikation festgestellt werden. Während die jüngeren Werker aus

Interesse an der Technik den Umgang mit Rechnern suchen, trifft man bei älteren Arbeitern oft auf verfestigte Verhaltensmuster, die den Einsatz neuer Technologien behindern.

Ein für die Werkstätten konzipiertes System muss in der Lage sein, flexibel verschiedene Benutzerprofile zu verwalten und entsprechend dieser Profile Informationen und Funktionalitäten zur Verfügung zu stellen.

Sowohl die Rohrwerkstatt als auch die Montagewerkstatt für Strukturelemente werden in drei Schichten betrieben. Vormittags- und Nachmittagsschicht bilden den Schwerpunkt der Arbeiten, während die Nachschicht als Pufferzone dient. Sind Termine eng gesteckt oder gibt es Probleme bei der Fertigung, wird die Nachschicht von halber auf volle Besetzung umgestellt, sodass auch sie dann als vollwertige Schicht betrachtet werden kann.

Der Schichtbetrieb beeinflusst den Informationsfluss zwischen den Werkern erheblich. Zwischen den Werkern der Schichten findet im Regelfall kaum Informationsaustausch statt. Es existiert lediglich ein kurzes Übergabegespräch zwischen den Vorarbeitern. Da die Auftragsplanung schichtübergreifend vollzogen wird, werden Aufträge über die Schichten hinweg auf verschiedene Werker verteilt, die aber aufgrund der zeitlichen Verschiebung keine Möglichkeit zur Kommunikation haben. Der Schichtbetrieb verhindert aufgrund der fehlenden Kommunikationsmöglichkeiten eine flexibel auf Probleme und wechselnde Anforderungen reagierende Fertigung.

Man kann festhalten, dass mit den bisherigen Mitteln, also sprachlicher Kommunikation, nur dann ein effektiver Informationsaustausch möglich ist, wenn ein ähnlicher Auftrag *gleichzeitig* an einer anderen Werkzelle bearbeitet wird. In allen anderen Fällen, beispielsweise bei der Bearbeitung eines Folgeauftrags in einer anderen Werkzelle, liegt eine zeitliche Differenz zwischen den zu kommunizierenden Ereignissen. Hier ist eine rein sprachgebundene Kommunikation nicht effektiv einzusetzen und der Einsatz von zeitlich ungebundenen Mitteln, z.B. E-Mail, erforderlich. Das Unterstützungssystem muss neue Kommunikationsformen durch entsprechende Funktionen fördern.

Ein weiterer Aspekt ist die strikte Trennung der Berufsgruppen Schiffbauer, Rohrschmied und Schweißer. Die horizontale Mobilität innerhalb dieses Systems, das heißt, die Möglichkeit des Wechsels innerhalb der Berufsgruppe ist größer, als die vertikale Durchlässigkeit, die einen Wechsel zwischen den Berufsgruppen bei entsprechender Qualifikation erlauben würde. Ein Wechsel innerhalb der Berufsgruppe in eine andere Werkstatt ist wahrscheinlicher, als ein Wechsel in eine andere Gruppe derselben Werkstatt, auch wenn eine entsprechende Qualifikation vorliegt.

Diese Trennung manifestiert sich auch in der werkerseitigen Organisation der Gruppen. Jeder Berufsgruppe steht ein spezieller Vorarbeiter vor (es gibt einen Schweißer-Vorarbeiter, einen Rohrschmied-Vorarbeiter und einen Schiffbauer-Vorarbeiter), der wiederum dem Vorarbeiter der Werkstatt untersteht. Die Trennung der Berufsgruppen verhindert den Austausch von Erfahrungen und eine weitere Qualifizierung der Werker. Eine effektive Zusammenarbeit der Mitglieder der verschiedenen Gruppen wird hierdurch erschwert. Positiver Effekt dieser Organisation ist die Spezialisierung der Gruppenmitglieder auf ihren Arbeitsbereich.

Große Unterschiede in der Qualifikation sowie die mangelnde horizontale Mobilität erschweren den Austausch von Erfahrungen und übergreifendem Wissen. Die Werker sind nur in ihrem speziellen Arbeitsgebiet einsetzbar und besitzen nur geringe Kenntnisse von Problemen und Erfordernissen anderer Fachgebiete. Schichtbetrieb und ein fehlendes Kommunikationssystem erschweren den Informationsaustausch der Werker. Flexibilität und Effizienz der Fertigung werden durch diese Faktoren nachhaltig verringert.

2.3 Heutige prozessunterstützende Systeme

In industriellen Werkstätten findet man bereits heute eine große Bandbreite informationstechnischer Geräte. Die Einführung rechnergestützter Maschinen legte in den 80er Jahren den Grundstein für den Einzug der Informationstechnik in die Werkstätten. Einfache CNC-Maschinen ermöglichen erste Automatisierungsschritte, später auch die digitale Einspeisung von Prozessdaten. Die Implementierung offener Schnittstellen der Maschinen leitete die nächste Phase der Datenintegration ein. Die neuen Schnittstellen ermöglichen die Integration von Rechnersystemen zur Aufzeichnung und Auswertung von Prozessparametern („Computer Integrated Manufacturing“, CIM).

Die zunehmend komplexer werdende Automatisierungstechnik führte zu einer weiteren Zunahme des Bedarfs an Informationstechnik in der Werkstatt. Maschinenspezifische Systeme werden heute zunehmend durch die PC-Technik abgelöst. Der rapide Preisverfall in diesem Bereich, die flexible Nutzbarkeit sowie die weitgehende Standardisierung sind nur einige Argumente, die für den Einsatz von PCs auf Werkstattebene sprechen.

Ein Nachteil der heute eingesetzten PC-Technik ist die unzureichende Eignung der Rechner für den Werkstatteinsatz. Gehäuse, Bildschirm und insbesondere die Eingabegeräte sind nicht gegen Schmutz und Feuchtigkeit geschützt und sind zum Beispiel für die Nutzung mit Handschuhen ungeeignet. Im Bereich der Nutzbarkeit

auf Werkstattebene sowie im Bereich der Ergonomie können erhebliche Defizite festgestellt werden.

Zudem sind PCs oft nicht an das betriebsweite Netzwerk angebunden, sondern stellen reine Einzelplatzlösungen dar. Eine durchgehende Integration von Konstruktions-, Planungs- und Auftragsdaten auf Werkstattebene wird dadurch verhindert.

Bei der Entwicklung des Unterstützungssystems für industrielle Werkstätten wurden zunächst verfügbare Produkte, aktuelle Entwicklungen und Trends untersucht. Es zeigte sich schnell, dass eine Vielzahl von Systemen existiert, die Teilbereiche des Konzeptes abdecken, jedoch keinen integrierten Lösungsansatz bieten.

Hinzu kommt, dass datentechnisch integrierte Informationssysteme zwar in vielen Betrieben verfügbar sind, diese jedoch aufgrund mangelnder Eignung nicht auf Werkstattebene eingesetzt werden können. So werden heute bereits in Verwaltung und Vertrieb häufig Groupwaresysteme eingesetzt, die die Zusammenarbeit und den Austausch von Informationen zwischen den Angestellten erleichtern. Diese Systeme bieten neben Kommunikationsfunktionen wie z.B. E-Mail auch Dokumentenmanagement-, Planungs- und Intranetdienste an. Beispiele für solche Systeme sind IBM Lotus Domino und Novell GroupWise. Diese Systeme sind ohne weitere Eigenentwicklungen wie z.B. die Implementierung einer zugeschnittenen Benutzeroberfläche aufgrund ihrer hohen Komplexität nicht für den Einsatz im Werkstattbereich geeignet.

In Werkstätten eingesetzte Softwaresysteme beschränken sich meist auf fertigungstechnisch motivierte Lösungen. Typische Beispiele für solche Systeme sind unter anderem Maschinensteuerungen und Produktionsdatenerfassungssysteme (CIM). Diese Systeme basieren auf rein technisch orientierten Ansätzen, das heißt, sie schaffen Teillösungen für spezielle technische Probleme. Sie stellen beispielsweise eine Bedienungsoberfläche für eine Maschine dar oder dienen der Auswertung der Produktionsleistung einer bestimmten automatisierten Arbeitszelle. Ihre Zielsetzung ist nicht die Unterstützung des Workers, nicht seine Motivation, nicht die Förderung einer Teamstruktur und damit letztlich nicht die zellenübergreifende Förderung von effizienter Arbeitsweise.

Im Bereich der Ressourcenplanung lässt sich ähnliches feststellen. Die in vielen industriellen Betrieben eingesetzten ERP-Systeme wie z.B. Baan IV und SAP/R3 bieten Planungsfunktionen auf hoher und mittlerer betrieblicher Ebene. Sie sehen jedoch keine Planungsfunktionen für Werkstätten vor bzw. bieten keine Möglichkeit der Neu- oder Umplanung aufgrund von kurzfristigen Ereignissen. Dieses Defizit wurde bereits von den Softwareherstellern erkannt und man plant in zukünftigen Versionen die Integration von entsprechenden Werkzeugen, die zwar eine

Werkstattplanung vorsehen, jedoch nicht für den direkten Einsatz in der Werkstatt konzipiert sind.

Die in heutigen Werkstätten eingesetzten Softwaresysteme stellen keine integrierte bzw. integrierende Lösung dar. Im Gegenteil: Der auf Teilespekte fokussierte Ansatz verhindert oftmals den Einsatz von neuen, innovativeren Werkzeugen, da durch die Teillösungen bereits PC-Arbeitsplätze innerhalb der Werkstatt belegt werden und faktisch als Negativbeispiel für die Integration von Informationssystemen in die Fertigungsumgebung gelten.

In den folgenden Abschnitten werden Unterstützungssysteme der Werkstätten untersucht und Defizite aufgeführt.

2.3.1 Rohrwerkstatt

In der Rohrwerkstatt werden punktuell einfache Informationssysteme eingesetzt. Ein typisches Beispiel ist in Abbildung 2-12 dargestellt. Es zeigt das Bedienpult einer



Abbildung 2-12: Installation eines Informationssystems an einer Rohrbiegemaschine

CNC-Rohrbiegemaschine. Die Steuerung der Maschine ist an einen PC gekoppelt, der über das Netzwerk mit dem Produktionsmanagementsystem PMS kommuniziert. Aus diesem System bezieht er das zur einem Auftrag gehörige CNC-Programm und überträgt es in die Steuerung. Der Auftrag selbst liegt dem Werker lediglich in Papierform vor. Er gibt die Auftragsnummer manuell in den PC ein. Eine Möglichkeit, weiterführende Informationen zum jeweiligen Auftrag oder zur Produktionsplanung zu erhalten, existiert nicht. Des Weiteren ist das System nicht an andere Arbeitszellen oder Datenbanken angebunden und stellt somit eine Insellösung dar, die ein koordiniertes Arbeiten im Team erschwert.

Im Bereich der Rohrwerkstatt werden zwar weitere Rechnersysteme eingesetzt, jedoch

werden diese nicht direkt durch den Werker, sondern durch den Vorarbeiter bedient, der zu Schichtbeginn die notwendigen Auftragsinformationen ausdrückt und an die Werker verteilt. Ein Beispiel für eine solche Lösung ist das Berechnungssystem für den Zuschnitt von Rohren. Das System berechnet auf Basis der vorliegenden

Produktionsaufträge die Längen der in der nächsten Zeit benötigten Rohrabschnitte. Diese Rohrabschnitte werden dann mithilfe eines Optimierungsverfahrens so auf das zur Verfügung stehende Rohmaterial verteilt, dass ein minimaler Verschnitt entsteht.

In diesen Beispielen spiegelt sich die zuvor beschriebene Situation wider. Rein technisch orientierte Teillösungen ohne integrativen Charakter bestimmen den zur Zeit sehr eingeschränkten Einsatz von rechnerbasierten Informationssystemen der Rohrwerkstatt.

2.3.2 Montagewerkstatt für Strukturelemente

Diese Werkstatt ist stärker automatisiert als die Rohrwerkstatt. Die Montage der Stahlprofile und Bleche wird zu großen Teilen automatisch vorgenommen. Entsprechend dem höheren Automatisierungsgrad der Werkstatt werden hier bereits komplexere Informationssysteme eingesetzt.

Als Beispiel seien hier Planungsmodul, Materialwirtschaftssystem und Steuerung der roboterbasierten Schweißanlage genannt. Diese Systeme bilden ähnlich wie in der Rohrwerkstatt innerhalb der Werkstatt Insellösungen, die nicht miteinander, sondern nur mit den darüber einzuordnenden werksweiten Systemen kommunizieren. Ihre Leistungsfähigkeit ist im Vergleich zu den Systemen der Rohrwerkstatt höher einzuordnen, da z.B. der Fortschritt der Schweißaufträge an der Roboterstation beobachtet werden kann. Diese Information kann dann zur manuellen Anpassung der Planungsdaten dienen.

Man erkennt, dass diese – wiederum rein technisch orientierten – Systeme lokale Probleme lösen, ohne das gesamte Werkstattumfeld und die Integration der anderen vorhandenen Systeme zu betrachten. Die eingesetzten Informationssysteme repräsentieren lediglich einen ersten Schritt in Richtung der weiteren Integration der Informationstechnik in industriellen Werkstätten.

3 Grundlagen der Kommunikation

3.1 Dezentrale Organisation

Durch die in den letzten Jahren aufgetretenen Veränderungen der Fertigungsstrukturen und die ständig wachsende Technisierung haben sich die Anforderungen im Bereich der Fertigungsorganisation stark gewandelt. Während wirtschaftlich motivierte Verbesserungen durch stärkere Automatisierung in weiten Bereichen ausgereizt und lediglich noch in Teilbereichen sinnvoll erscheinen, verlagert sich der Schwerpunkt des Rationalisierungspotenzials mehr und mehr in Richtung der Optimierung der Fertigungsorganisation [JRO94].

Die industrielle Produktion war in der Vergangenheit – und ist heute noch – in vielen Bereichen gekennzeichnet durch eine starre Hierarchie, auf deren unterster Ebene der Arbeiter in der Werkstatt eingeordnet ist. Der Arbeiter gilt in dieser Sichtweise als stochastisches Element, das einer größtmöglichen Kontrolle unterzogen werden muss, um Abweichungen vom Produktionsplan zu vermeiden [IMA97].

Die zunehmende Globalisierung der Märkte, kurze Produktlebenszyklen, eine zunehmende Zahl von Vermarktungskanälen sowie eine Veränderung der Kostenstruktur erfordern eine Umstrukturierung der Betriebe auf Produktionsebene.

In den folgenden Abschnitten wird ein Konzept zur dezentralen Produktion vorgestellt, dessen Ziel die Steigerung der Produktionseffizienz durch eine stärkere Partizipation der Werker und eine Flexibilisierung der Organisationsstrukturen ist.

Dabei geht der Trend weg von starren Hierarchien, die im heutigen Marktumfeld nicht genügend Flexibilität mitbringen und hin zu flachen, dynamischen Organisationsformen, die größere Spielräume für eine adaptive, kundenorientierte Produktion bieten. Kunden können hier sowohl Kunden des Unternehmens als auch im industriellen Umfeld nachfolgende Werkstätten sein, die mit Teilprodukten beliefert werden.

In der Literatur findet man eine Vielzahl von Ansätzen, deren Ziel die Reorganisation des Unternehmens oder eines Teiles des Unternehmens ist. Da der Fokus dieser Arbeit auf der Optimierung von Produktionsprozessen liegt, sollen hier nur solche Ansätze vorgestellt werden, die diesen Randbedingungen genügen. Gleichzeitig soll das Konzept realisierbar, das heißt, kontinuierlich implementierbar sein, da kurzfristige Änderungen und eine abrupte Umstellung von Organisationsstrukturen zu Produktionsausfällen führen kann [FRE98].

Im Bereich der Computer Supported Cooperative Work (CSCW) finden sich umfangreiche Ansätze zur Effizienzsteigerung im industriellen Umfeld. Unter den Begriffen Business Process Reengineering (BPR) und Workflow-Management (WFM) werden Lösungen diskutiert, deren gemeinsamer Kern die Schaffung von partizipativen Arbeitsumgebungen und damit einer flachen Hierarchie bzw. dynamischen Gruppenstruktur ist [MON97].

Alle Ansätze haben gemeinsam, dass sie zur Unterstützung der angestrebten Reorganisation den umfangreichen Einsatz von Informationstechnologie voraussetzen [NAH95], [WET96], [TSM95], [CZR97]. Die Informationstechnik stellt hier neue Kommunikationsmechanismen für die Gruppenarbeit sowie Systeme zur Unterstützung von Entscheidungen (DSS, Decision Support Systems) zur Verfügung [STV94].

Zur erfolgreichen Realisierung einer Umstrukturierung im Sinne einer dezentralen Produktion muss ein Paradigmenwechsel stattfinden, der die Förderung des Einsatzes der menschlichen Fähigkeiten zum Ziel hat [IMA97], [JOH98]. Um den Wettbewerbsvorteil dieser neuen Paradigmen zu nutzen und voranzutreiben, muss das Unternehmen über eine Struktur verfügen, die sich durch folgende Charakteristika auszeichnet:

→ Reaktionsschnelligkeit

- Schnell reagierende Elemente der Unternehmensstruktur
- Aufgabenbezogene, nicht hierarchieabhängige Kommunikation
- Motivierende Arbeitsumgebung und Identifikation mit den Zielen des Unternehmens
- Minimale Totzeiten durch flexible Planung und Werkerpartizipation

→ Zielgerichtetes Handeln

- Feinere Auftragsgranulierung mit enger Zielsetzung
- Interne Optimierung auf Basis von marktwirtschaftlichen Kriterien
- Risiko- und Gewinnteilung sorgen für gemeinsame Ziele von Management- und Ausführungsebene

➔ Flexibilität

- Hohe Reaktionsfähigkeit des Unternehmens durch konkurrierende Geschäftseinheiten, die selbständig neue Märkte suchen
- Rollen statt Funktionen als zentrales Element der Aufgabenverteilung
- Geförderte Eigenverantwortlichkeit der Individuen für Teilziele

➔ Kooperationsbereitschaft

- Jede Unternehmenseinheit kann als Kompetenzzentrum dienen und Wissen intern und extern zur Verfügung stellen
- Identifikation mit dem Unternehmen durch konfliktfreie Arbeitsumgebung
- Das Management steuert Elementrelationen anstelle von Elementen und sucht kontinuierlich nach neuen potenziellen Relationen
- Entwicklungskosten und -ergebnisse können im Netzwerk aus Unternehmenseinheiten aufgeteilt werden

3.1.1 Holonisches Modell

Ein Ansatz, der in den letzten Jahren großes Interesse sowohl in der Forschung als auch in der Industrie geweckt hat, ist das sogenannte holonische Fertigungssystem, das viele der oben genannten Anforderungen erfüllt [WHS94], [GUB98]. Beim holonischen Fertigungssystem wird jeder partizipierende Teil des Fertigungsprozesses als Individuum betrachtet, das bestimmte Charakteristika erfüllt. Diese Individuen werden nach [KOE89] als „Holen“ bezeichnet.

Der Begriff „Holon“ setzt sich aus dem griechischen „holos“ (ganz, gesamt) und dem Suffix „on“ (Partikel, Teilchen) zusammen. Er drückt aus, dass ein Holon einerseits eine autonome Einheit bildet und andererseits im Zusammenhang mit der gesamten Fertigung steht. Die Aufgabe eines Holons besteht darin, Objekte, zu denen auch Informationen gezählt werden, zu transformieren. Diese Transformation besteht z.B. im Verarbeiten, Montieren, Verpacken oder Transportieren der Objekte.

Das Modell der holonischen Fertigung basiert auf einem Baukasten elementarer Holontypen, die – zu einem Ganzen zusammengefügt – das Fertigungssystem ergeben. Das in Abbildung 3-1 dargestellte System basiert auf den drei als Basis dienenden Holontypen

- ➔ Auftragsholon,
- ➔ Produktholon und
- ➔ Ressourcenholon.

Alle Holonentypen können grundsätzlich miteinander interagieren, also Informationen austauschen oder im Netzwerk Aufgaben lösen.

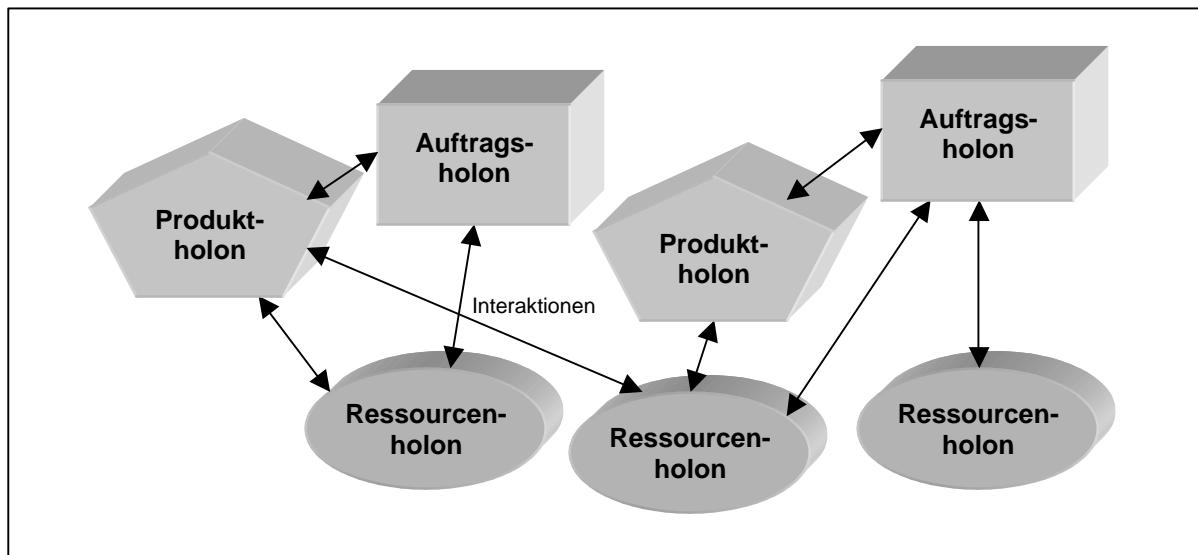


Abbildung 3-1: Beispiel für ein holonisches Fertigungssystem

Holenen besitzen grundsätzlich die Eigenschaften der Autonomie und Kooperativität. Unter Autonomie wird die Fähigkeit zur Eigenständigkeit, also der selbständigen Planung und Ausführung von Prozessen verstanden. Kooperativität ist die Fähigkeit, diese Aktionen im Team auszuführen [LÜT98].

Ein Holon besteht immer aus einem informationsverarbeitenden Teil und zusätzlich oft aus einem physikalischen, prozessverarbeitenden Teil. Ein Holon kann Teil eines anderen Holons sein.

Das holonische Fertigungssystem integriert die gesamte Bandbreite der Fertigungsprozesse von der Auftragsvergabe über die Konstruktion, die Produktion und den Vertrieb und bildet so ein flexibles Unternehmen in einem Modell ab.

Die Fertigkeiten des Menschen sind eine wesentliche Komponente des holonischen Fertigungssystems. Umgekehrt helfen holonische Konzepte dem Menschen, ihre Fähigkeiten zu bündeln und zielgerichtet einzusetzen [BUS98], [BON98].

Das holonische Fertigungssystem stellt die Methoden für ein anwendungsorientiertes Konzept zur Verfügung, das im folgenden Kapitel beschrieben wird.

3.1.2 Übertragung auf den Schiffbau

Das holonische Fertigungssystem liefert eine theoretische Basis, anhand derer die Analyse einer bestehenden Fertigungsstruktur vollzogen werden kann. Mithilfe dieser Basis werden die Grundelemente der Unternehmensstruktur identifizierbar und

können somit als Grundlage für die Gestaltung einer neuen, flexibleren und effizienteren Struktur herangezogen werden.

Die spezifischen Anforderungen der Fertigung im Schiffbau, die durch eine hohe Produktvarianz, eine geringe Losgröße und eine hohe Werkstückkomplexität gekennzeichnet sind, macht in Kombination mit stark inhomogenen Qualifikationsprofilen der Werker die Einführung einer Teamstruktur erforderlich.

Das hier vorgestellte Konzept bildet die angestrebte Fertigungsorganisation mithilfe der zuvor beschriebenen Methoden auf die Merkmale der teamorientierten Arbeitsweise ab. Ein Team besteht dabei aus einer Anzahl von Werkern, die durch ihre Zusammenarbeit Synergieeffekte, die sich aus unterschiedlicher Erfahrung, Ausbildung usw. ergeben, nutzen. Die inhomogene Zusammensetzung der Teams stellt dabei sicher, dass das Team autonom unterschiedliche Aufträge aus seinem Arbeitsbereich annehmen und ausführen kann. Da die Interaktion von Mitgliedern innerhalb eines Teams sowie von Teams untereinander der entscheidende Faktor der Leistungsfähigkeit der Teamstruktur ist, muss das holonische Modell der Teamarbeit um die Basisholonen

- ➔ Kommunikation und
- ➔ Information

erweitert werden. Diese beiden neuen Holontypen stellen hauptsächlich Verbindungen zwischen Ressourcen-, Auftrags- und Produktholonen her. Die konsequente Fortführung des holonischen Modells der Fertigung in Teams impliziert des weiteren die Einführung eines Teamholons, das andere Holonen beinhalten kann. Die Einführung des Teamholons ermöglicht die Betrachtung von Interaktionen eines Teams mit anderen Holonen. Eine Interaktion kann sowohl teamintern – zwischen den Mitgliedern eines Teams – als auch teamextern erfolgen. Teamexterne Interaktion beinhaltet die Interaktion zwischen zwei Teams, die Interaktion zwischen einem Team und einem beliebigen anderen Holon außerhalb des Teams sowie die Interaktion eines Teammitglieds mit einem Mitglied eines anderen Teams. Ein Beispiel für ein holonisches Modell einer teamorientierten Fertigung ist in Abbildung 3-2 dargestellt.

Das Teamholon könnte hier z.B. aus zwei Werkern und einer Maschine, also drei Ressourchenholonen, einem Auftrag, einer Kommunikationseinrichtung wie z.B. einem Email-System sowie einem Informationsholon, z.B. Erfahrungswissen oder einer Datenbank, bestehen. Das Team und seine Mitglieder interagieren während der Bearbeitung des Auftrags mit weiteren Holonen, hält Rücksprache mit anderen Teams, akquiriert externe Informationen oder vergleicht die aktuelle Arbeit mit ähnlichen Aufträgen.

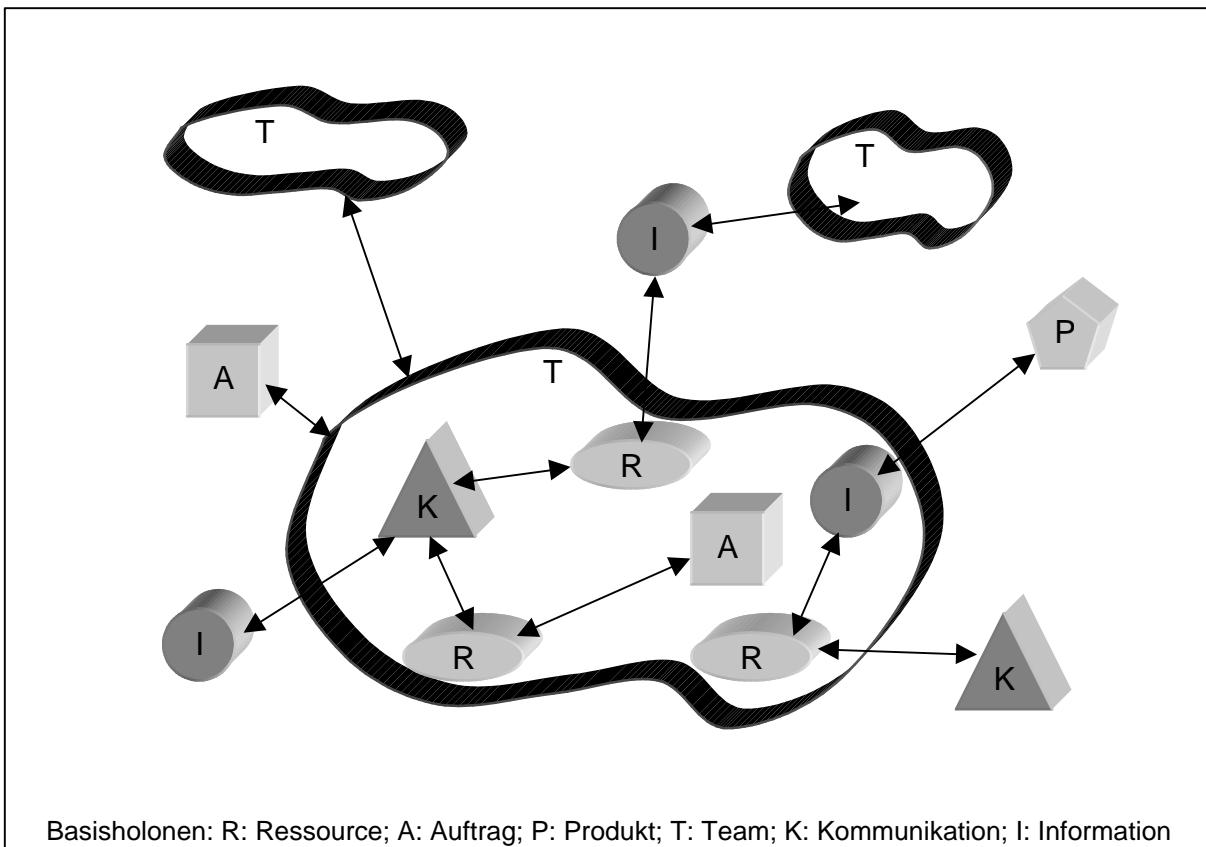


Abbildung 3-2: Holonisches Modell teamorientierter Fertigung

Die Effizienz der neuen Teamstruktur basiert auf der Ausbildung eines Netzwerkes aus Informationsaustausch und Kommunikation zwischen den Basisholonen Ressource, Auftrag und Produkt. Ohne die Komponenten dieses Netzwerks wäre die Teamstruktur starr und nicht in der Lage, sich flexibel an neue Situationen anzupassen oder sich Wissen anzueignen. Erst Informationsaustausch und Kommunikation setzen die Synergieeffekte der Teamstruktur frei. Sie sorgen für die Anhäufung und gleichzeitig den Austausch von Expertenwissen innerhalb und außerhalb eines Teams und fördern so konsequent die Nutzung des Potenzials der Werker. Das Team stellt damit eine autonom und flexibel agierende Einheit des Fertigungsprozesses dar.

Es wird deutlich, dass Information und Kommunikation die Schlüsselkomponenten einer erfolgreichen Reorganisation der Werkstätten darstellen. Nur mit ihrer Hilfe verspricht die Einführung einer flachen Hierarchie auf Basis einer Teamstruktur Erfolg. Die mit der Kommunikation einhergehende Redundanz soll dabei durch eine gezielte Analyse der Anforderungen sowie durch die Verwendung eines anwenderorientierten Systemdesigns gemindert werden.

Das Modell der holonischen Fertigung stellt die theoretische Basis für die in den Werkstätten angestrebte Reorganisation dar. Aus diesem Modell folgen direkt die

Forderungen nach kommunikationsunterstützenden Maßnahmen. In den folgenden Abschnitten werden zunächst die zu fördernden Kommunikationskanäle untersucht, bevor im Anschluss die Definition der Rahmenanforderungen an das Unterstützungssystem folgt.

3.2 Kommunikation

3.2.1 Horizontale Kommunikation

Horizontale Kommunikation bedeutet Kommunikation innerhalb der Fertigungsumgebung, in diesem Fall also meist Kommunikation innerhalb der Werkstatt. Wie Abbildung 3-3 zeigt, findet horizontale Kommunikation nicht nur zwischen verschiedenen Stationen eines Fertigungsprozesses, sondern auch zwischen verschiedenen Prozessen statt.

So können bei dem Weg eines Werkstückes durch die Werkstatt Informationen mit dem Werkstück weitergegeben werden oder später im Fertigungsprozess folgende Stationen können Rückfragen an vorhergehende stellen. Diese Kommunikationsform ist besonders dann wichtig, wenn Probleme oder Fehler an einer Station aufgetreten sind, die dem Folgeprozess mitgeteilt werden müssen. Fehler können auch erst im Folgeprozess auffallen, sodass hier eine Rückmeldung an den Vorgängerprozess erfolgen muss.

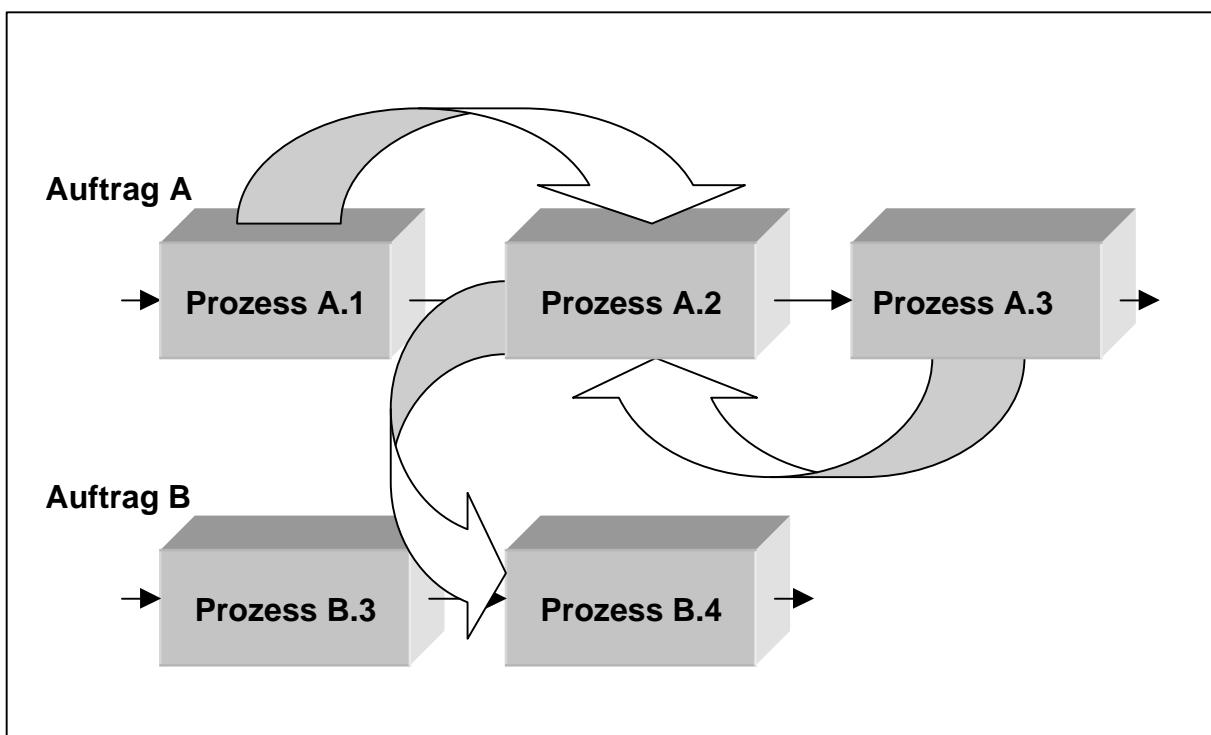


Abbildung 3-3: Horizontale Kommunikation

Zusätzlich zur Kommunikation entlang eines Auftrages ist auch die Kommunikation zwischen unterschiedlichen Aufträgen sinnvoll. Der Informationsaustausch zwischen ähnlichen Aufträgen führt zu einer Optimierung der Fertigung und direkten Reduktion der Fehlerrate.

3.2.2 Vertikale Kommunikation

Als vertikale Kommunikation (siehe Abbildung 3-4) wird der Informationsaustausch zwischen Werkstatt und den in der Unternehmenshierarchie darüber liegenden Abteilungen bezeichnet.

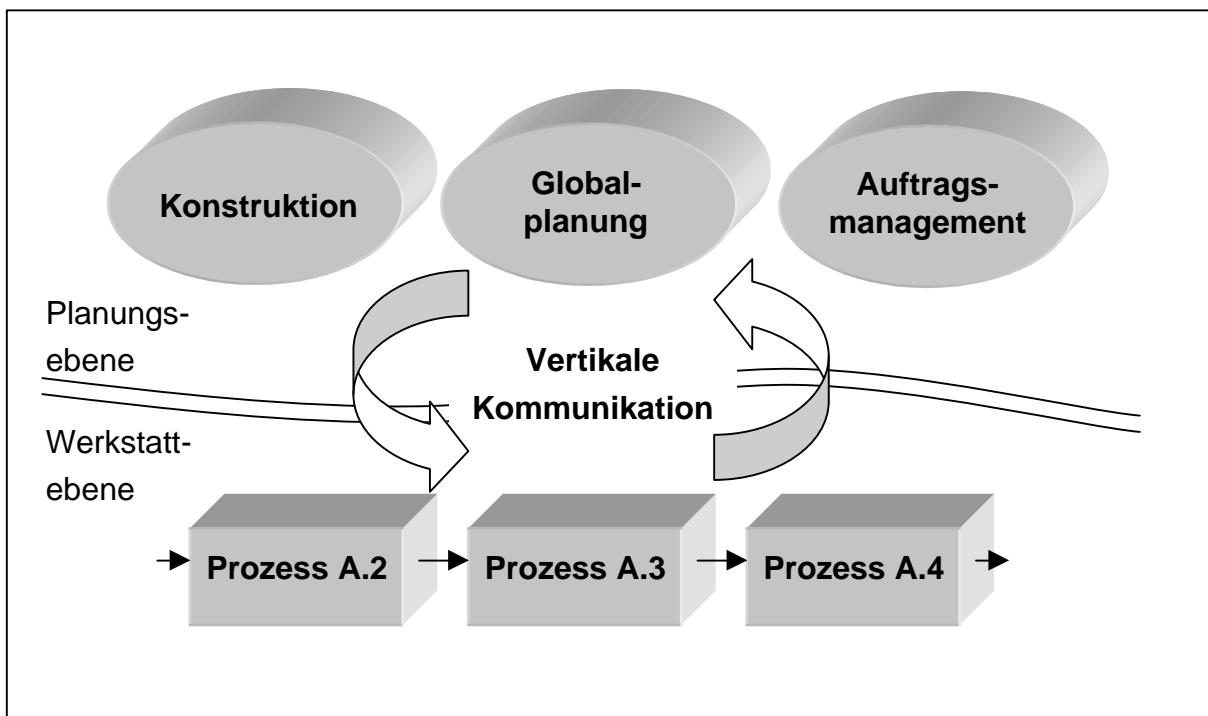


Abbildung 3-4: Vertikale Kommunikation

Diese Kommunikationsform sorgt für eine Rückkopplung von Informationen bezüglich Konstruktion, Planung und Auftragsmanagement aus der Werkstatt. In dem so entstehenden Regelkreis könnten zum Beispiel Verbesserungsvorschläge sofort der Konstruktionsabteilung mitgeteilt werden.

3.3 Systemanforderungen

Die Anforderungen an ein benutzerorientiertes prozessunterstützendes System für den Einsatz in industriellen Fertigungsumgebungen lassen sich in inhaltliche, technische, funktionale und ökonomische Anforderungen gliedern. In den folgenden Abschnitten werden diese Anforderungen beschrieben.

3.3.1 Inhaltliche Anforderungen

Als inhaltliche Anforderungen werden solche Anforderungen verstanden, die sich auf Art und Umfang der im Unterstützungssystem dem Anwender zur Verfügung gestellten Informationen und Funktionen beziehen. Die Inhalte des Unterstützungssystems sollen die Arbeit in Teams fördern, den Werker über aktuelle Zustände des Fertigungssystems informieren und eine umfassende Bibliothek an Informationsmaterial verschiedenster Art enthalten.

Ein wichtiger Aspekt der Informationsvermittlung ist die benutzerorientierte Komposition der Inhalte. Das System muss in der Lage sein, Informationen benutzerbezogen zur Verfügung zu stellen. Der Zugriff auf bestimmte Informationen kann so auf entsprechende Anwendergruppen oder einzelne Anwender beschränkt werden. Beispielsweise sollen nur Mitglieder der Gruppe „Administratoren“ in der Lage sein, Veränderungen, etwa die Verwaltung von Benutzerdaten, am System vornehmen zu können. Ist ein anderer Anwender eingeloggt, dürfen die entsprechenden Funktionen nicht im Menü angezeigt werden. Neben der Systemsicherheit kann mithilfe dieser Technik gewährleistet werden, dass der Anwender nur die für ihn in der aktuellen Situation relevanten Informationen zur Auswahl hat und er nicht durch eine Vielzahl von für ihn überflüssigen Informationen irritiert wird [HAS97].

Die Inhalte des Unterstützungssystems sollen unerfahrene Werker und Auszubildende in die Lage versetzen, auch komplexe Arbeiten selbstständig ausüben zu können. Unter anderem müssen Anweisungen zur Ausführung schwieriger Arbeiten und zur Bedienung und Wartung komplexer Maschinen zur Verfügung gestellt werden. Weitere wichtige Informationen sind beispielsweise Fertigungs- und Qualitätsstandards, Sicherheitsanweisungen und Unfallstatistiken. Das System soll so Aus- und Weiterbildung fördern und die Chancengleichheit der Werker verbessern. Gleichzeitig soll es als zentrale Sammelstelle für Expertenwissen dienen, die Synergieeffekte und die Verbreitung von Erfahrungen fördert [FIS98].

Erfahrungsaustausch und die Verbesserung der Teamstruktur stehen auch im Vordergrund der Forderung nach horizontaler und vertikaler Kommunikation. Das System muss Möglichkeiten zum persönlichen und öffentlichen Austausch von Informationen zur Verfügung stellen. Die Kommunikation muss unabhängig von zeitlichen und räumlichen Einflussfaktoren erfolgen können. So sollen zum Beispiel die Werker der nachfolgenden Schicht über die Ereignisse der laufenden Schicht informiert werden, oder Informationen über Probleme oder Fehler in der Konstruktion sollen von den Konstrukteuren selbst nachgelesen werden können. Zusätzliche Kommunikationsfunktionen sollen den Werker über allgemeine Ereignisse wie Termine, Veranstaltungen oder Fortbildungen informieren.

Durch die Kopplung des Systems mit den Werkstückdatenbanken und Produktionsplanungssystemen des Betriebs soll der Werker in die Lage versetzt werden, ausführliche Informationen zum Werkstück, seinen aktuellen Zustand sowie Lagerort einholen zu können. Eine anschauliche und interaktive Visualisierung des Werkstücks muss zur Verfügung gestellt werden.

Die Anbindung der Produktionsplanungssysteme sowie die Erweiterung ihrer Funktionalität soll die Planbarkeit und Reaktionsfähigkeit der Planung auf aktuelle Ereignisse verbessern. Die Aufträge müssen auf Werkstattebene interaktiv vom Werker bzw. vom Team planbar und anpassbar sein, um Flexibilität und Effizienz zu steigern.

Die Inhalte sind in statische, dynamische, interne und externe Informationen zu gliedern. Statische sowie systeminterne Informationen werden vom System selbst gespeichert und können direkt vor Ort administriert werden. Dynamische bzw. externe Informationen erfordern die Integration des Systems in die datentechnische Infrastruktur des Betriebes.

3.3.2 Technische Anforderungen

Um die möglichst nahtlose Integration des Unterstützungssystems in das technische Umfeld der Werkstatt zu gewährleisten, muss es verschiedene technische Anforderungen erfüllen.

Grundsätzlich soll die Entwicklung des Unterstützungssystems unter Verwendung aktuellster Technologien, die heute bereits etablierter Standard sind oder deren Entwicklung zum industriellen Standard absehbar sind, erfolgen. Dadurch soll sichergestellt werden, dass zukünftige Installationen nicht zwingend an neue Technologien angepasst werden müssen. Gleichzeitig soll das auf diesen Technologien basierende Unterstützungssystem als „Leuchtturmapplikation“ dienen, die als Motivation für den verstärkten Einsatz von aktuellen Informationstechnologien auf Werkstattebene fungiert.

Das System muss einfach an neue Umgebungen und Anforderungen angepasst werden können. Die Basis des Systems bleibt dabei stets gleich, es werden lediglich Teilmodule angepasst, aus einem Modulbaukasten hinzugefügt oder ersetzt. Diese modulare Architektur garantiert gleichzeitig die für zukünftige Anpassungen benötigte Skalierbarkeit des Systems. Die zum Einsatz kommenden Entwicklungsplattformen sollen soweit wie möglich auf objektorientierten Ansätzen basieren, um spätere Anpassungen zu vereinfachen. Die objektorientierte Modulstruktur ermöglicht dabei die Wiederverwendung von Klassen, deren Methoden grundlegende Funktionalitäten zur Verfügung stellen.

Für die Plattform des Systems soll auf standardisierte, weit verbreitete Systeme zurückgegriffen werden. Dies sichert ebenso wie die Architektur sowohl die Adaptivität als auch die Akzeptanz des Systems. Vorhandene Hard- und Software kann so weiterverwendet werden und eventuell vorhandene Erfahrungen der Anwender können so eingebracht werden.

Die Orientierung an aktuellen und zukünftigen Standards steht im Vordergrund der Konzipierung der Werkzeuge. Aktuelle, internetbasierte Netzwerktechnologie, Datenaustausch-Standards wie ODBC, CORBA als Middlewareplattform und innovative Datenformate sollen untersucht und auf ihre Einsetzbarkeit geprüft werden.

3.3.3 Ergonomische Anforderungen

Das Unterstützungssystem muss neben den inhaltlichen und technischen Anforderungen vor allem auch den ergonomischen Anforderungen genügen. Die ergonomischen Anforderungen an das Unterstützungssystem umfassen die Anforderungen an die grafische Benutzeroberfläche (Graphical User Interface, GUI) sowie die Anforderungen an die zum Einsatz kommenden Rechnersysteme und Peripheriegeräte.

Die Entwicklung eines Mensch-Rechner-Systems erfordert stets die vorherige Analyse der potenziellen Benutzer des Systems. Bedienbarkeit und Akzeptanz müssen unter Berücksichtigung der späteren Zielgruppe sichergestellt werden können.

Neben den Einflüssen der persönlichen Merkmale der Nutzer muss zusätzlich das soziale Umfeld der Werkstätten untersucht werden. Das System soll mit dem sozialen Umfeld des Workers harmonieren, es darf nicht gegen tradierte Regeln der Arbeitsorganisation verstößen [SKL97].

3.3.3.1 Ergonomie der grafischen Benutzeroberfläche

Wesentlich für die Entwicklung eines benutzerorientierten Unterstützungssystems ist die ergonomische Gestaltung der grafischen Oberfläche [SPI87]. Da das System prozessnah, das heißt auf Werkstattebene zum Einsatz kommen soll, muss dabei beachtet werden, dass die Bedienung durch Worker erfolgt, die bisher keinen oder nur wenig Kontakt mit Rechnersystemen hatten.

Die Grundsätze der ergonomischen Dialoggestaltung wurden bereits 1988 als Norm in der DIN 66234, Teil 8 [DIN66] sowie in ISO 9241, Teil 10 verankert und in [BSW93] bewertet:

Aufgabenangemessenheit

Das Softwaresystem insgesamt muss in seiner Architektur so konzipiert werden, dass es den Benutzer auf wichtige Kerninformationen aufmerksam macht, ohne ihn mit weniger wichtigen Informationen zu überfordern. Des weiteren ist darauf zu achten, dass das System einfach und eindeutig in seiner Bedienung ist, jedoch Zusatzinformationen in verschiedenen Detailgraden jederzeit angefordert werden können, falls die Situation es erfordert.

Selbstbeschreibungsfähigkeit

Soweit möglich, sind für die jeweiligen Aufgaben Funktionen vorzusehen, die den Anwender von einem Arbeitsschritt zum nächsten begleiten, ohne jedoch eine zu starre Benutzerführung vorzugeben. Zusätzlich müssen stets zu allen Systemzuständen Hilfestellungen verfügbar sein.

Beispiel: Zu einem Schweißprozess soll eine schrittweise Anleitung geliefert werden können, die eindeutig den Sprung zum nächsten, zum vorherigen sowie zum ersten Schritt erlaubt. Zusätzlich zu den prozessbezogenen Hilfestellungen werden systembezogene Informationen angeboten.

Steuerbarkeit

Der Anwender muss zu jedem Zeitpunkt vollständige Kontrolle über das System haben. Er muss selber in der Lage sein, seine Arbeitsgeschwindigkeit zu bestimmen und Einfluss auf den Ablauf der Informationsschritte zu haben. Der Anwender soll geleitet werden, jedoch ohne dass ihm eine bestimmte Vorgehensweise aufgezwungen wird.

Beispielsweise müssen in allen Dialogen Rücksprung- bzw. Abbruchtasten deutlich sichtbar angebracht sein, um die Möglichkeit anbieten zu können, schnell in einen dem Anwender bekannten Systemzustand zu wechseln.

Erwartungskonformität

Die Erwartungskonformität soll durch eine einheitlich gestaltete, sich vorhersehbar verhaltende Benutzeroberfläche sichergestellt werden. Der Benutzer soll dadurch in die Lage versetzt werden, sich ein Regelwerk des Systemverhaltens aufzubauen. Die Erwartungskonformität bezieht sich nicht nur auf das optische Erscheinungsbild der Benutzeroberfläche, sondern auch auf Systemstruktur und -reaktionen.

Beispiel: Rot dargestellte Informationen weisen immer auf eine kritische Aktion hin; graue Tasten führen den Anwender durch einzelne Arbeitsschritte; vor dem Löschen einer Datei erscheint immer eine Sicherheitsabfrage, etc.

Fehlerrobustheit

Das Unterstützungssystem soll in der Lage sein, fehlerhafte Eingaben zu tolerieren oder Korrekturvorschläge zu machen. Fehlerhafte Eingaben sind unter anderem

- ➔ Zahlenwerte außerhalb eines vordefinierten Gültigkeitsbereichs,
- ➔ Formatfehler, z.B. ungültige Zeichen, Buchstaben statt Zahlen,
- ➔ fehlende Eingaben und
- ➔ strukturelle bzw. inhaltliche Fehler, die sich aus dem Zusammenhang ergeben.

Soweit möglich, sollen manuelle Eingabefelder nur gültige Zahlen bzw. Zeichen erlauben. Ist die Gültigkeit nicht bereits während der Eingabe prüfbar, muss das System nach der Wertübergabe und -prüfung den Fehlerzustand eindeutig kennzeichnen und eine entsprechende Ausgabe einer Fehlermeldung veranlassen. Fehlende Eingaben sollen dadurch ausgeschlossen werden, dass das Verlassen einer Eingabemaske nur dann möglich ist, wenn alle notwendigen Felder ausgefüllt sind. Der Anwender ist dabei auf den fehlenden Wert hinzuweisen.

Strukturelle bzw. sich aus dem Zusammenhang ergebende Fehler sind nicht sofort bei der Eingabe zu erkennen und stellen sich erst in einer späteren Analyse der Eingabedaten heraus. Tritt ein solcher Fehler auf, ist der Benutzer durch eine eindeutige Fehlermeldung darauf hinzuweisen. Die Fehlermeldung muss dabei das Problem möglichst genau beschreiben und soll Vorschläge zur Korrektur des Fehlers enthalten oder darauf hinweisen, dass eine automatische Korrektur erfolgt ist.

Lernförderlichkeit

Das System soll das Erlernen neuer Fähigkeiten mit Bezug sowohl auf den Fertigungsprozess als auch auf die Handhabung des Unterstützungssystems fördern. Angepasst an die Fähigkeiten und Kenntnisse des Benutzers soll es Verbesserungsvorschläge und Alternativlösungen anbieten und erklären. Dabei soll besonders auf spezifische Vorteile einer alternativen Lösung hingewiesen werden.

Individualisierbarkeit

Individualisierbarkeit ist eine der wichtigsten Voraussetzungen für ein benutzerorientiertes Unterstützungssystem. In Abhängigkeit von personenspezifischen Faktoren wie Ausbildung, Fähigkeit, mit dem System umzugehen und Kenntnis der verschiedenen Fertigungsprozesse muss das Unterstützungssystem unterschiedliche Inhalte und Detailtiefen darstellen können. Zusätzlich sind benutzerabhängig Konfigurationsmöglichkeiten für die Oberfläche

vorzusehen. Je nach Nutzerqualifikation sind diese Einstellungen vom Anwender selbst oder vom Administrator vorzunehmen.

Zum Beispiel müssen Menüfunktionen in Abhängigkeit vom Benutzer konfigurierbar sein, sodass einzelne Funktionen oder Funktionsgruppen nur für bestimmte Anwender sichtbar sind.

3.3.3.2 Einsatz von Metaphern

Zum Verständnis komplexer Systeme verwenden Menschen häufig sogenannte „mentale Modelle“, die während des Erlernens eines Systems oder bei seiner Benutzung schrittweise aufgebaut werden. Die Gliederung des mentalen Modells in Teilmodelle erlaubt dem Menschen, das Gesamtsystem in einfacher zu verstehende Teilsysteme zu zerlegen. Ziel der Gliederung ist die Aufteilung in genügend kleine Teilaufgaben, die mit bereits bekannten oder auch trivialen Handlungsstrukturen gelöst werden können.

Für die erfolgreiche Nutzung des Rechners ist neben der Konsistenz, Klarheit und Problemangemessenheit des realisierten Systems vor allem die Kompatibilität des mentalen Modells des Benutzers und des konzeptuellen Modells des Systemdesigners entscheidend. Jede Unstimmigkeit führt im geringsten Fall zu erhöhten Beanspruchungen des Benutzers, im schlimmsten Fall zur Unbenutzbarkeit des Systems. Es ist daher unbedingt sicherzustellen, dass kompatible Modelle entstehen können.

Abstrakte Systeme, in diesem Fall ein komplexes Softwaresystem, müssen daher so konzipiert werden, dass der Anwender die globale Aufgabe („Bedienung des Gesamtsystems“) in Teilaufgaben („Bedienung einer Funktion“) zerlegen und auf bekannte Strukturen abbilden kann.

Die Bildung eines mentalen Modells kann durch den Einsatz von Metaphern stark vereinfacht werden. Als Metapher bezeichnet man eine rhetorische Figur, die das Gemeinte durch eine Vorstellung zum Ausdruck bringt, die aus einem anderen Bereich stammt und keine reale Beziehung zum Gemeinten hat. Eine Metapher ist ein bekanntes kognitives Modell, das eine partielle Kongruenz mit einer neu zu erlernenden kognitiven Struktur aufweist. Diese partielle Kongruenz erlaubt die Interpretation des zunächst unbekannten und somit unverständlichen Systemverhaltens.

Die Auswahl geeigneter, möglichst unmissverständlicher Metaphern, die den zukünftigen Benutzern den Aufbau eines leistungsfähigen mentalen Modells vom System erlauben, sollte am Anfang jedes Oberflächen- und Dialogentwurfs stehen.

Für Metaphern gibt es zahlreiche Beispiele. Sehr verbreitet ist die „Desktop-metapher“, die die Funktionen einer Software mit den symbolisch dargestellten Elementen eines Schreibtisches verdeutlicht [BBH97].

3.3.3.3 Anordnung von Information

Neben der Beachtung von inhaltlich-repräsentativen Gestaltungsmitteln sind grafische Abbildung und Layout der Dialoge zu untersuchen. Das menschliche Gedächtnis hat die Eigenschaft, zusätzlich zur vermittelten Information auch den Ort des Auftretens dieser Information zu speichern. Struktur und Anordnung von Informationen können dazu dienen, die Wiederauffindbarkeit zu erhöhen und Wissen besonders einprägsam zu vermitteln.

Als maßgebend für die Strukturierung bzw. Ordnung von Wissen auf dem Bildschirm können vor allem Nähe, Symmetrie und Gleichartigkeit von Informationen angesehen werden.

Als wichtige Gestaltungsaspekte werden von [HER94] verschiedene Aspekte erachtet. *Einfache Formen* erhöhen danach die Prägnanz. Einfachheit erweckt den Eindruck von Deutlichkeit und Klarheit. Bei nicht geordneter Komplexität entstehen dagegen Irritation und Unsicherheit. Modelle, die nach dem Prinzip der *Regelmäßigkeit* aufgebaut sind, werden ästhetisch höher eingestuft und gegenüber regellosen, chaotischen Formen bevorzugt. Regelmäßigkeit erhöht die Verständlichkeit und Überschaubarkeit eines Systems. *Geschlossene Formen* wirken figurbildend, ausgleichend und harmonisierend. Eine *symmetrische Gestalt* ermöglicht durch ihr leicht erfassbares Prinzip und ihren einprägsamen Charakter eine einfache und schnelle Orientierung. Symmetrie bewirkt Ausgewogenheit, Stabilität und die ausgeprägteste Art des Gleichgewichts. *Asymmetrie* erzeugt dagegen Reiz und Spannung. Die *Gleichartigkeit* von ähnlichen oder zusammengehörenden Informationen sowie komplementär dazu die *Andersartigkeit* der Repräsentation unterschiedlicher Informationen sind weitere wichtige Grundsätze.

Aus diesen Grundsätzen folgen Richtlinien für die Wahl des richtigen Layouts:

- Logisch zusammenhängende Informationen sollen auch örtlich zusammen aufgeführt werden.
- Ähnliche Information soll auch ähnlich dargestellt werden, z.B. mit gleichem Font, gleicher graphischer Gestaltung (Farbe, Umrandung, etc.).
- Unterschiedliche Information soll durch klare Konturen voneinander abgesetzt sein, z.B. durch klar ausgerichtete Zwischenräume, Trennlinien, Ausrichtung nach linker und rechter Seite.

- Gleiche bzw. ähnliche Information auf unterschiedlichen Formularen (bzw. Bildschirmseiten) soll immer an der gleichen Stelle des jeweiligen Formulars auftreten (Ortscodierung, Erwartungskonformität).
- Die Information soll prinzipiell von links oben nach rechts unten nach ihrer Wichtigkeit und der Häufigkeit ihrer Verwendung angeordnet werden (Leserichtung).
- Geläufige, prägnante Begriffe, gleiche Syntax sowie Groß- und Kleinbuchstaben erleichtern die Aufnahme von Informationen. Abkürzungen sind soweit möglich zu vermeiden.

3.3.3.4 Fehlermeldungen

Kommt es zu unerwarteten Eingaben, die erst im Zusammenhang mit weiteren Informationen einen Fehlerzustand ergeben, muss das System angemessen mit einer Fehlermeldung reagieren. Nach [FRZ91] sind die Kriterien für die ergonomische Gestaltung von Fehlermeldungen in informationale und motivationale Aspekte zu unterteilen.

Die informationalen Aspekte beinhalten, dass eine Fehlermeldung gut sichtbar, informativ, verständlich und orientierend sein soll. Ein prägnantes Design soll den Fehlerzustand signalisieren. Gleichzeitig sollen dem Anwender Informationen zum aktuellen Systemzustand, der letzten Aktion sowie dem vermutlichen Fehler mitgeteilt werden. Die Verständlichkeit wird durch eine kurze, konkrete Beschreibung des Fehlers unter Berücksichtigung des Vorwissens des Anwenders erhöht. Als Hilfestellung sind dem Anwender Orientierungshilfen zur Fehlerbehandlung zu geben.

Als motivationale Aspekte bezeichnet [FRZ91] die Forderung nach Höflichkeit und nach Vermeidung von Redundanz. Die Meldung soll demnach keine beleidigenden Formulierungen, die etwa dem Benutzer Fehler vorwerfen, beinhalten, Belehrungen vermeiden und keine Vermutung über die Intentionen des Benutzers anstellen. Insgesamt soll der Umfang der Fehlermeldungen konfigurierbar sein.

Die oben genannten Kriterien sollen bei der Konzipierung des Systems berücksichtigt werden. Angemessene Fehlermeldungen sind eine wichtige Voraussetzung für die Sicherstellung der Erlernbarkeit der Software.

3.3.3.5 Hardware-Ergonomie

Neben der Ergonomie der Software muss auch auf die Ergonomie und Benutzbarkeit der zum Einsatz kommenden Hardware geachtet werden.

Ergonomische Gesichtspunkte sind unter anderem Mobilität des Gerätes, Qualität, Lichtstärke und Größe der Anzeige und Ergonomie der Eingabegeräte wie Maus, Tastatur, Stift oder Joystick.

Zusätzlich zu diesen Gesichtspunkten muss die Verwendbarkeit im Werkstattbereich berücksichtigt werden. Maßgebliche Faktoren sind Erschütterungsempfindlichkeit, Schmutz- und Feuchtigkeitsresistenz, Bedienbarkeit der Eingabegeräte mit Arbeitshandschuhen sowie die elektromagnetische Verträglichkeit des Rechners und seiner Peripheriegeräte.

An Arbeitsstationen mit großen Bauteilen bzw. Maschinen oder im Lager muss der Anwender das Unterstützungssystem vor Ort nutzen können. In diesem Fall kann der Einsatz eines portablen PCs sinnvoll sein. Zwar gibt es eine große Bandbreite an verfügbaren Geräten, jedoch müssen meist Nachteile wie eine kleine oder lichtschwache Anzeige oder ein empfindliches Gehäuse in Kauf genommen werden. Eine Untersuchung typischer Vertreter mobiler Rechnersysteme soll zeigen, inwiefern heutige Systeme den Anforderungen der Werkstatt gewachsen sind.

Die Leistung der hier untersuchten mobilen Systeme ist teilweise mit denen marktüblicher stationärer PCs vergleichbar, jedoch muss hier mit weitaus höheren Anschaffungs- und Erweiterungskosten gerechnet werden.

Als erstes tragbares System wurde ein Fujitsu Stylistic 2300 (Abbildung 3-5 links)



Abbildung 3-5: Fujitsu Stylistic 2300 (l.) [FUJ00], Panasonic Toughbook 71 (r.) [PAN00]

getestet, dessen Leistungsfähigkeit mit der moderner PC-Systeme vergleichbar ist. Sein selbstleuchtendes Farbdisplay und das drahtlose Netzwerk zeichnen ihn aus. Die Bedienung erfolgt mithilfe eines Stiftes, der nicht nur die Funktion einer Maus erfüllt, sondern auch zur Texteingabe verwendet werden kann. Die in diesem Gerät verwendete Festplatte ist durch eine Gel-Lagerung gegen Stöße geschützt. In Verbindung mit einer robusten, gepolsterten Schutztasche, die die Bedienung wenig beeinflusst, ist dieses Gerät in Teilbereichen des Werkstattumfelds, z.B. im Lager,

einsetzbar. Für den Einsatz direkt an Arbeitsstationen ist das Gerät aufgrund der empfindlichen Anzeige ungeeignet.

Ein wesentlich robusteres Gerät ist das „Toughbook 71“ von Panasonic (Abbildung 3-5 rechts). Hierbei handelt es sich um einen Laptop-PC, der für den Einsatz in rauen Umgebungen konzipiert wurde. Er verfügt über ein spritzwasser- und staubdichtes Magnesium-Gehäuse, ein integriertes Funknetz sowie über eine stoßfest gelagerte Festplatte. Auch hier ist der Einsatz direkt in der Werkzelle nur bedingt denkbar. Zwar ist der Laptop robuster als der Fujitsu Stylistic, jedoch können Display und Tastatur noch als Schwachpunkte gesehen werden. Zudem ist die Bedienbarkeit im mobilen Einsatz sehr eingeschränkt, da das Gerät zur Benutzung aufgeklappt und auf eine ebene Unterlage gestellt werden muss. Als geeignet erscheint das Gerät nur für die Verwendung bei der Planung, bei Transportaufgaben und Lagerarbeiten.

Zusammenfassend muss festgestellt werden, dass zwar eine Vielfalt an tragbaren, leistungsfähigen PC-Systemen verfügbar ist, ihre Verwendbarkeit im Umfeld der Produktion jedoch nur eingeschränkt möglich erscheint. Zwar sind diese Systeme oft technisch hervorragend ausgestattet, jedoch unter mechanischen Aspekten nicht ausreichend geschützt. Als weiterer Nachteil ist der hohe Kostenaufwand für ein solches System zu nennen, der häufig um den Faktor drei bis vier über dem eines herkömmlichen PCs liegt.

Stationäre PCs sind aufgrund stabilerer Gehäuse besser geschützt und sind nicht der Gefahr des Herunterfallens ausgesetzt.



Abbildung 3-6: Stationäre PC für den Werkstatteinsetz [PRO00]

Die in Abbildung 3-6 gezeigten Geräte der Firma Protech sind zwar stationär montiert, können jedoch mit Schwenkarmen versehen werden, sodass sie auch als Informationsstation zwischen mehreren Werkzellen fungieren können.

Die sehr stabilen, wasser- und schmutzdichten Gehäuse und die gute Erweiterbarkeit der Systeme sind weitere positive Aspekte. Es kann festgehalten werden, dass stationäre, entsprechend ausgerüstete PCs heute uneingeschränkt für das Unterstützungssystem verwendet werden können. Ist man auf mobile Lösungen angewiesen, müssen Einschränkungen im Einsatzbereich hingenommen werden.

3.3.4 Ökonomische Anforderungen

Neben den beschriebenen Erfordernissen dürfen wirtschaftliche Gesichtspunkte nicht außer acht gelassen werden. Wichtiges Kriterium bei der Einführung neuer Technologien ist die Höhe der Investitionskosten, die durch zu erzielende Einsparungen gerechtfertigt sein muss. Die Einsparungen sollen durch die mithilfe des Unterstützungssystems zu erzielende Effizienzsteigerung der Fertigung sowie die höhere Motivation der Werker erreicht werden.

Die bei der Einführung des Unterstützungssystems entstehenden Gesamtkosten lassen sich im wesentlichen in drei Kategorien, die direkt aus den Hauptphasen eines solchen Projekts ersichtlich sind, einordnen.

Diese Kategorien sind

- Kosten für eine Analyse der Fertigung und des resultierenden Optimierungspotenzials,
- Kosten für Implementierung und Installation und
- Kosten für Systempflege und Wartung.

In der Vorphase der Entwicklung muss zunächst das Optimierungspotenzial der Werkstatt auf Basis einer Analyse der Fertigungsorganisation identifiziert werden. Die hier zu erwartenden Kosten hängen hauptsächlich von der Größe der Werkstatt und vom Komplexitätsgrad der Fertigung ab.

Im Anschluss an die Analyse erfolgt die Entwicklung der Software sowie die Beschaffung und Integration der notwendigen informationstechnischen Infrastrukturen wie z.B. PC-Hardware und Netzwerktechnik. Eine erste Schulung der Mitarbeiter muss vorgesehen werden.

Zusätzliche laufende Kosten für Mitarbeiterschulungen und Folgekosten für Systemwartung und Erweiterungen müssen in Betracht gezogen werden.

Es muss zwischen systembedingten und externen Kostenfaktoren unterschieden werden. Externe Kostenfaktoren liegen in der jeweiligen Zielwerkstatt begründet und

sind nicht vom Systemdesign abhängig. Systembedingt sind Kosten, die durch das Konzept und die Gestaltung des Systems anfallen. Neben den Implementierungskosten sind dies Kosten für Hardware, zusätzliche Software, Installation, Schulungsmaßnahmen sowie Kosten, die durch aufwendige spätere Aktualisierungen und Erweiterungen anfallen. Diese Kosten können durch das Konzept und die Architektur des Systems maßgeblich beeinflusst werden.

Durch die Auswahl einer verbreiteten, leicht zu bedienenden Softwareplattform soll erreicht werden, dass möglichst viele Anwender bereits Erfahrungen besitzen und dadurch geringere Schulungskosten anfallen. Durch die Verwendung von standardisierten Komponenten bei der Auswahl der Hardwareplattform sowie beim Softwaredesign soll sichergestellt werden, dass eine spätere Wartung oder Erweiterung des Systems vereinfacht und damit verbilligt wird. Die Modularität der Architektur soll dies unterstreichen.

4 Technologische Plattform

Im folgenden Abschnitt wird auf die Kriterien und Auswahl der technologischen Plattform eingegangen. Eine Auswahl soll nach unten aufgeführten Gesichtspunkten erfolgen:

- ➔ Ergonomie,
- ➔ Kosten und
- ➔ Einsetzbarkeit im Werkstattumfeld.

Es ist grundsätzlich von drei verschiedenen Aufgabenstellungen für die technologische Plattform auszugehen:

- ➔ Kostengünstige, gleichzeitig einfach zu bedienende Systeme für die Benutzeroberfläche (Client),
- ➔ leistungsfähige Serverdienste für die Akquisition und Aufbereitung von Daten sowie ein
- ➔ echtzeitfähiges Subsystem für die Akquisition von Maschinen- und Prozessdaten.

Die Betriebssysteme stellen fundamentale Dienste wie Netzwerkanbindung, grafische Benutzeroberfläche und Dateisysteme zur Verfügung.

Das Unterstützungssystem soll so weit wie möglich auf bereits in den Werkstätten existierende Informationstechnik aufsetzen.

4.1 Betriebssysteme

Die in industriellen Werkstätten eingesetzte Software umspannt aufgrund der unterschiedlichen Aufgabenstellungen und Rechnerausstattungen einen weiten Bereich.

Abhängig von der jeweiligen Aufgabe und ihren jeweiligen Anforderungen findet man verschiedene Betriebssysteme bzw. Betriebssystemgruppen, die in Abbildung 4-1 dargestellt sind.

Maschinennahe Systeme wie Speicherprogrammierbare Steuerungen (SPS), Mikrocontroller- oder komplexere Echtzeitsysteme werden für Steuerungsaufgaben und Echtzeit-Datenverbindungen zu anderen Werkzellen verwendet. Diese Systeme sind jedoch meist nicht für eine Interaktion mit dem Anwender konzipiert und werden ausschließlich von Experten bedient.

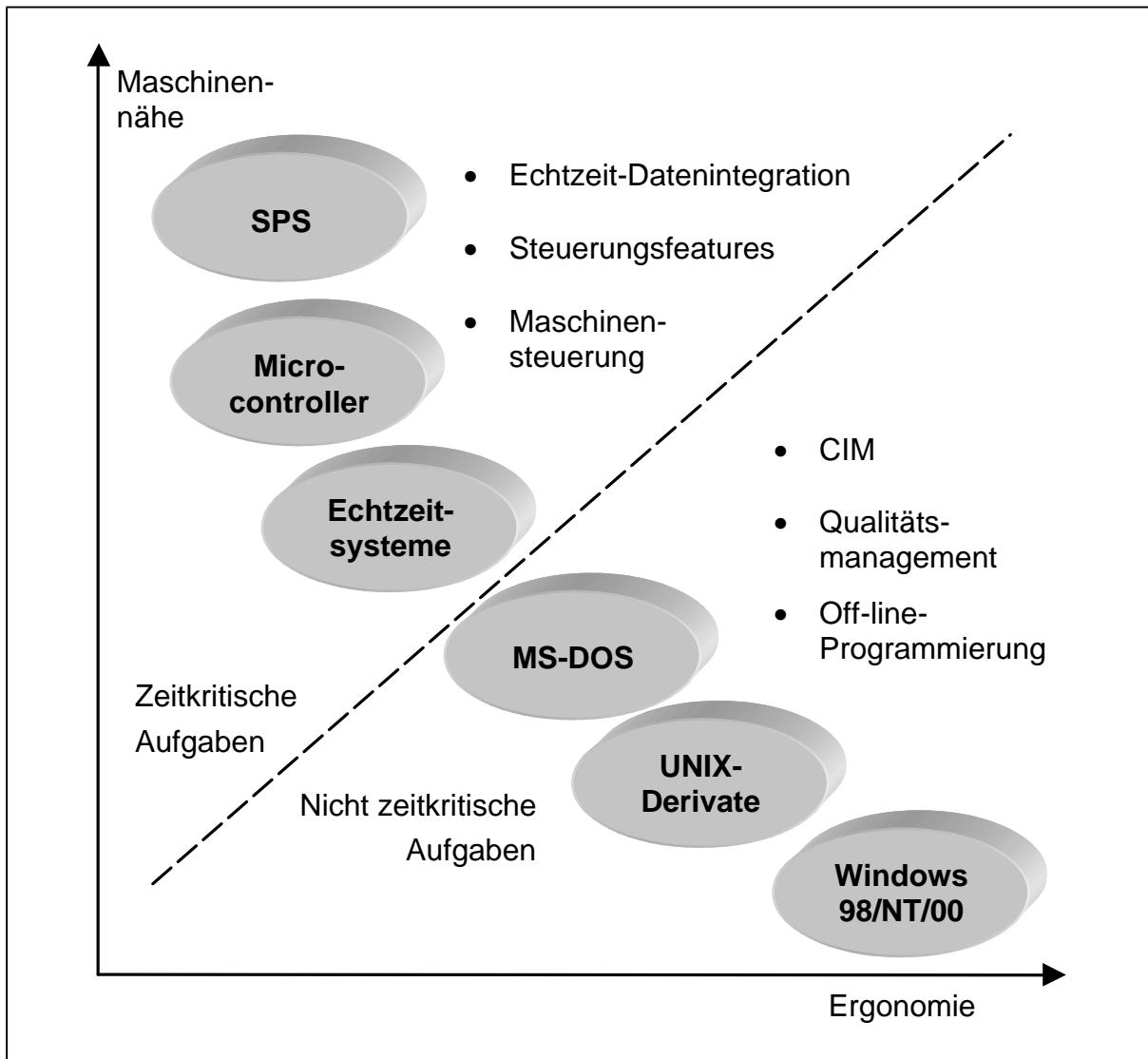


Abbildung 4-1: Betriebssysteme auf Werkstattebene

Erfordert die Aufgabenstellung eine Benutzerinteraktion, so wird oft auf MS-DOS oder MS-Windows zurückgegriffen. Seltener findet man Derivate der UNIX-Familie, die meist für Speziallösungen verwendet werden. Diese Systeme sind nicht echtzeitfähig, jedoch sind sie relativ einfach zu handhaben. Insbesondere die MS-Windows-Familie ist sehr verbreitet, sodass man hier oft auf Vorwissen der Werker trifft.

Aufgrund der unterschiedlichen Anforderungen und der Trennung der Module aus Echtzeit- und Sicherheitsgründen ist eine Aufgabentrennung zwischen Client, Server und Echtzeitsystem notwendig. Tabelle 4-1 zeigt die verschiedenen Anforderungen auf.

	Echtzeitfähigkeit	Netzwerk-integration	Ökonomie	Ergonomie
Client	gering	mittel	hoch	hoch
Server	mittel	hoch	gering	mittel
Echtzeitsystem	hoch	mittel	gering	gering

Tabelle 4-1: Anforderungsprofil der Systemplattformen

Client und Server müssen nur geringen bzw. mittleren Anforderungen im Bereich der Echtzeitfähigkeit genügen. Sie dienen im wesentlichen zur Visualisierung und Übermittlung von Daten. Zeitkritische Funktionen sind dabei in das Echtzeitsystem ausgelagert und durch entsprechend konzipierte Kommunikationsmechanismen entkoppelt. Im Bereich der Netzwerkintegration werden die höchsten Anforderungen an den Server des Unterstützungssystems gestellt. Dieser akquiriert Daten verschiedener Quellen mithilfe verschiedener Protokolle aus dem Netzwerk und muss gleichzeitig alle Verbindungen zu den aktuell aktiven Clients aufrechterhalten. Die Netzwerkanforderungen an Client und Echtzeitsystem sind weniger komplex, da sie jeweils nur eine Verbindung zum Server unterhalten müssen.

Es wird davon ausgegangen, dass eine typische Installation des Unterstützungssystems aus einem zentralen Server, einer festen Anzahl von echtzeitfähigen Maschinenanbindungen sowie einer größeren variablen Zahl von Clients besteht. Daraus ergeben sich direkt die in der Tabelle dargestellten ökonomischen Kriterien, die die jeweiligen Systeme erfüllen sollen.

Der Client stellt die Schnittstelle zum Werker dar und stellt die höchsten Anforderungen an die Ergonomie, da die Bedienung intuitiv und ohne weitreichende Vorkenntnisse möglich sein muss. Der Server wird im Regelfall nur von ausgebildeten Administratoren bedient, sodass er in Bezug auf ergonomische Gesichtspunkte weniger kritisch ist. Direkte Interaktionen mit dem Echtzeitsystem erfolgen ausschließlich durch Entwickler zur Installation, Konfiguration oder Wartung. Die Ergonomie der Echtzeitplattform spielt daher hier nur eine untergeordnete Rolle.

4.1.1 Client

Auf der Seite des Clients überwiegen die Forderungen nach Ergonomie und Ökonomie. Seine Echtzeitfähigkeit spielt nur eine untergeordnete Rolle.

Stichprobenartige Umfragen in den Werkstätten haben ergeben, dass – sofern Erfahrungen mit Rechnern vorliegen – diese in über 90% der Fälle mit dem Betriebssystem Microsoft Windows 95/98 gemacht wurden. Für eine realitätsnahe 3D-Darstellung von CAD-Daten kann auf den verbreiteten Standard OpenGL zurückgegriffen werden, der in das System integriert ist. Zudem ist das Betriebssystem auf handelsüblichen PCs lauffähig und somit kostengünstig einsetzbar. Durch die Integration des flexibel einsetzbaren TCP/IP Protokolls ist das Betriebssystem für den Einsatz als Client ausreichend netzwerkfähig.

4.1.2 Server

Der Server als zentrale Komponente des Unterstützungssystems erfordert eine hohe Zuverlässigkeit sowie eine leistungsfähige Netzwerkanbindung.

Als verhältnismäßig einfach zu wartende Lösung hat sich Microsoft Windows NT in vielen Betrieben bewährt. Windows NT zeichnet sich durch eine hohe Systemstabilität sowie eine integrierte Benutzerverwaltung aus. Der Zugriff auf das Dateisystem des Servers kann benutzer- und gruppenspezifisch konfiguriert werden.

Ein anderes, ähnlich weit verbreitetes Serverbetriebssystem ist Novell NetWare. Da der in dieser Arbeit zum Einsatz kommende Intranetserver Lotus Domino jedoch nicht zu Novell NetWare kompatibel ist, soll Windows NT als Server für das Unterstützungssystem eingesetzt werden. Durch die Möglichkeit der Integration der Nutzerverwaltung von Windows NT in Lotus Domino wird die Administration des Servers vereinfacht.

4.1.3 Echtzeitsystem

Nach [DIN44] handelt es sich bei einem Echtzeitsystem um ein System, das anfallende Daten derart verarbeitet, dass die Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeitspanne verfügbar sind. Ein solches Verhalten ist bei der Kopplung an einen Fertigungsprozess notwendig, da der Prozess durch Parameterabfragen o. ä. in seinem Zeitverhalten nicht oder nur klar definiert beeinflusst werden darf. Verschiedene Prozessparameter sind stark zeitabhängig, sodass die aufgenommenen Messwerte mit präzisen Zeitstempeln versehen werden müssen, um später sinnvoll analysiert werden zu können. Prozessüberwachungs-, Steuerungs- und Eingriffsmöglichkeiten erfordern zudem ein sehr schnell reagierendes, mit einer Vielzahl von Schnittstellen ausgerüstetes Betriebssystem [HÜS94].

Durch die echtzeitfähige Bereitstellung der Information kann von Seiten des Benutzers sowie der integrierten Prozessüberwachung und -regelung in laufende Prozesse schnell und effektiv eingegriffen werden. Diese selektive Bereitstellung von

Information und die Erzeugung von Zusatzinformationen in Kombination mit flexiblen Eingriffen in den laufenden Prozess ermöglichen somit eine optimale Kontrolle der Fertigungseinrichtung durch den Maschinenbenutzer [BEN99].

Als Integrationsplattform für zeitkritische Anwendungen innerhalb des Unterstützungssystems ist ein POSIX-kompatibles und damit ein einem weit verbreiteten Standard entsprechendes Betriebssystem gewählt worden. Die Plattform basiert auf dem Echtzeit-Betriebssystem VxWorks 5.2. Die Hardwarearchitektur besteht aus drei voneinander unabhängig agierenden PowerPC-Prozessoren, die über den VME-Bus über einen gemeinsamen Speicherzugriff verfügen. Das System verfügt so über ein hohes Maß an Skalierbarkeit und kann flexibel an verschiedene Umgebungen angepasst werden.

4.2 Interface-Technologien und Datenformate

Das verteilte Unterstützungssystem basiert auf einer Client-Server-Architektur. Die Synthese der dem Benutzer übermittelten Informationen wird in Abhängigkeit von der jeweiligen Situation und den Benutzereigenschaften auf der Seite des Servers vorgenommen. Die so aufbereitete Information wird dann anhand von Layoutspezifikationen formatiert und an den Client über das Netzwerk übermittelt.

4.3 Intranet

Ein Intranet ist ein organisationsweit eingesetztes Netzwerk, das mit weiteren Netzwerken inner- und außerhalb der Organisation verbunden sein kann. Eine Verbindung zum Internet ist möglich und wird oft mit Hilfe von Gateways hergestellt. Das Intranet bedient sich meist der Technologien des Internet, also der TCP/IP-Protokollfamilie und Web-Browsern als Benutzerschnittstelle [EHM97], [EVA96].

Durch die Verwendung von Intranettechnologien kann das verteilte Benutzerunterstützungssystem vorhandene Infrastrukturen nutzen [WKG97], [ZIV98]. Die Synthese bzw. die Integration der im Unternehmen netzwerkweit verfügbaren Daten mit Daten aus der Produktion führt zu einem Synergieeffekt, der die Entwicklung von neuen Funktionen und Bewertungsmechanismen auf Basis der Kombination der Informationen ermöglicht [BEU98], [BUD97].

Ein weiteres Argument für den Einsatz von Intranettechnologien ist die hervorragende Portierbarkeit und Integrierbarkeit in ein heterogenes Netzwerk mit verschiedenen Hard- und Softwareplattformen, wie es im Unterstützungssystem eingesetzt wird [WSH98].

Beim Einsatz eines Web-Browsers als Applikation für die Benutzerinteraktion kann man zudem auf Vorkenntnisse der Anwender aus anderen Intranetapplikationen bzw. aus der Nutzung des Internet aufbauen.

4.4 Aktive Module

Neben der Präsentation von auf Clientseite statischen Inhalten wie zum Beispiel eines Handbuches oder einer zweidimensionalen Grafik muss das System auch clientbezogen aktive Unterstützungsmodule integrieren.

Aktive Module dienen der Visualisierung, Überwachung und Simulation des Fertigungsprozesses. Dabei soll der Anwender vor, während und nach dem Fertigungsprozess durch die aktiven Module unterstützt werden. Weniger erfahrene Werker können so selbständiger arbeiten und sich interaktiv Wissen aneignen.

Als aktive Module sollen folgende Funktionalitäten für das Unterstützungssystem implementiert werden:

- ➔ Echtzeitnahe Visualisierung und Überwachung von Prozessparametern,
- ➔ echtzeitnahe Simulation und Verifikation des Prozesses,
- ➔ interaktive 3D-Visualisierung von Werkstücken,
- ➔ werkstattorientierte Aufgabenplanung und
- ➔ Offline-Simulation des Fertigungsprozesses.

Für die Implementierung dieser Module reichen die bisher beschriebenen dokumentorientierten Methoden nicht aus. Es sollen daher die im Folgenden beschriebenen Technologien zum Einsatz kommen. Dabei soll berücksichtigt werden, dass die Module in die intranetbasierte Systemarchitektur integrierbar sein müssen.

4.4.1 Java

Experten halten Java für einen zukünftigen Standard für aktive, inter- und intranetbasierte Module in industriellen Umgebungen [ATH98], [CON98]. Die Programmiersprache Java ist für die verteilte Anwendung in Netzwerken konzipiert und ähnelt in ihrer Syntax und Struktur der verbreiteten Sprache C++. Mit Java entwickelte Anwendungen können auf drei unterschiedliche Weisen genutzt werden:

- ➔ **Application:** Ein Java-Programm läuft auf einem einzelnen Rechner ab und verhält sich ähnlich wie herkömmlich entwickelte Software.

- ➔ **Applet.** Das Java-Programm befindet sich auf einem Server und wird vom Client angefordert. Nach der Übertragung wird es innerhalb des Clients ausgeführt. Es kann weiterhin mit dem Server kommunizieren, um zum Beispiel Daten aus einer zentralen Datenbank anzufordern.
- ➔ **Servlet.** Das Java-Programm befindet sich auf einem Server und übernimmt dort serverspezifische Aufgaben wie zum Beispiel die Beantwortung von Datenbank-Abfragen.

Java unterstützt ausschließlich eine objektorientierte Programmierung. Die Wiederverwendbarkeit von Klassen verkürzt die Entwicklungszeit und fördert die Modularisierung der Programme. Ein weiterer Vorteil von Java ist die hohe Portabilität. Java-Programme sind grundsätzlich ohne erneute Kompilierung auf unterschiedlichen Plattformen lauffähig und sind so leicht in Netzwerke aus heterogenen Systemplattformen zu integrieren [JEN97].

Für die Entwicklung der aktiven Module für die Visualisierung von Prozessparametern und für kleinere Planungsaufgaben und Auftragsverwaltung stellt Java eine ideale Lösung dar. Module können mit Java nahtlos in die Benutzeroberfläche integriert werden und dort mit dem Anwender interagieren. Auch die Integration von Datenbanken über standardisierte Schnittstellen ist in JAVA vorgesehen [HAO97], [STE00].

Für die Implementierung von komplexen Applikationen ist Java aufgrund des technischen Grundkonzeptes und der daraus resultierenden geminderten Leistungsfähigkeit nur bedingt geeignet. Die Plattformunabhängigkeit muss durch Einbußen der Ausführungsgeschwindigkeit erkauft werden. Für komplexe oder performancekritische Applikationen wird oft die Sprache C++ verwendet. Nachteil dieser Sprache ist, dass die resultierenden Programme nicht plattformunabhängig sind, nicht innerhalb eines Web-Browsers ausgeführt werden können und damit nicht intranetfähig sind.

4.4.2 Tool Command Language / Toolkit (TCL/Tk)

Die „Tool Command Language“, im Folgenden kurz als TCL bezeichnet (sprich: „Tickle“), wurde von Prof. J. Ousterhout an der Universität Berkeley entwickelt und wird inzwischen von der Firma Scriptics kommerziell eingesetzt und vertrieben. Für Forschungszwecke ist TCL/Tk kostenlos nutzbar [OUS95].

TCL ist ähnlich strukturiert wie die Sprache Java und ist ebenso wie diese objektorientiert und plattformübergreifend einsetzbar. TCL ist eine leistungsfähige Skriptsprache, die zur Laufzeit von einem Interpreter verarbeitet wird. Somit ist nicht nach jeder Programmänderung ein erneuter Kompiliervorgang notwendig. Der TCL-

Interpreter, der in C++ entwickelt wurde, liegt vollständig im Quelltext vor und kann beliebig durch neue Funktionen, also neue Skriptbefehle, erweitert werden.

Das Zusatzpaket „Toolkit“ (kurz „Tk“) basiert auf der Sprache TCL und bietet die Möglichkeit, TCL-Programme mit einer grafischen Benutzeroberfläche zu versehen.

Für TCL/Tk kann ein Plugin für Web-Browser installiert werden. Damit ist das Ausführen von in dieser Sprache entwickelten Programmen innerhalb des Web-Browsers, ähnlich wie Java-Applets, möglich. TCL/Tk besitzt gegenüber Java den Vorteil, dass durch die Skriptingfähigkeit die Entwicklungszeit eigener Programme stark reduziert wird. Programme und neu entwickelte Befehle können sehr einfach innerhalb der TCL-Shell getestet werden. Durch die Möglichkeit, dem Interpreter eigene Befehle hinzuzufügen, können auch komplexe Aufrufe zum Beispiel der Grafikbibliothek OpenGL in die Skriptsprache integriert werden. Die eigentliche Ausführung der Befehle erfolgt aus dem Interpreter heraus, der komplett in C++ geschrieben ist. Somit sind TCL/Tk-Programme bei entsprechendem Aufbau, das heißt, bei Auslagerung von Instruktionen mit hohen Performanceanforderungen, wesentlich leistungsfähiger als entsprechende Programme in Java.

4.4.3 3D-Visualisierung und modellorientierte Prozessüberwachung

Die Arbeitsprozesse mit automatisierten oder teilautomatisierten Maschinen sind oft komplex und für den Werker unter zeitlichen und geometrischen Aspekten nur schwierig vorstellbar. Oft kann dies bei der Fertigung von geometrisch aufwendigen Werkstücken festgestellt werden.

Die Fertigung von geometrisch komplexen Werkstücken erfordert ein hohes Maß an räumlichem Vorstellungsvermögen und man erkennt deutlich den Bedarf an Lösungen zur Visualisierung der Fertigungsprozesse sowie der Zielprodukte. In den folgenden Abschnitten wird die technische Basis für ein Prozessmodellierungs- und überwachungswerkzeug sowie für ein Werkstückvisualisierungswerkzeug vorgestellt.

4.4.3.1 Werkstückvisualisierung

In beiden untersuchten Werkstätten werden ständig wechselnde, sich stark unterscheidende Werkstücke gefertigt. Der Werker muss sich bei jedem Auftrag erneut ein Bild der Werkstückgeometrie machen, um bereits vor dem Beginn des Fertigungsprozesses die Reihenfolge des Zusammenbaus und die Zahl und Art der vorzufertigenden Teilprodukte abschätzen zu können.

Ein Visualisierungsmodul würde hier die Möglichkeit bieten, die Effizienz der Arbeit zu steigern und die Zahl der Produktionsfehler nachhaltig zu reduzieren.

Da der offene Ansatz des Unterstützungssystems erhalten bleiben soll, muss das Visualisierungsmodul in der Lage sein, unterschiedliche Modellformate darstellen zu können.

Als besonders geeignet erscheint in diesem Bereich VRML (Virtual Reality Modelling Language). VRML ist eine Skriptsprache zur Beschreibung von dreidimensionalen Modellen und ihrer kinematischen Eigenschaften. Für viele Web-Browser können Plugins für die Anzeige von VRML-Modellen installiert werden. Somit können dreidimensionale Modelle direkt in das Unterstützungssystem integriert werden. Die Darstellung der dreidimensionalen Objekte ist interaktiv und ermöglicht zum Beispiel Zoom und Drehung des Modells. Da viele VRML-Plugins zur Darstellung der Grafik OpenGL verwenden, wird die Leistung von 3D-Beschleunigerkarten genutzt.

4.4.3.2 Prozessvisualisierung

Fertigungsprozesse mit automatisierten Maschinen sollen durch eine 3D-Visualisierung online und offline simuliert werden können.

Als Prozessbeispiel wird in dieser Arbeit das Schweißen mit einem sechsachsigen Industrieroboter herangezogen. Grundsätzlich soll das Werkzeug jedoch so konzipiert werden, dass auch andere Maschinen, zum Beispiel eine Rohrbiegemaschine oder eine CNC-Fräsmaschine modelliert werden können.

Für die Realisierung der 3D-Simulation soll OpenGL verwendet werden. OpenGL (Open Graphics Library) bietet eine offene, sehr leistungsfähige Plattform für die Entwicklung von dreidimensionalen Visualisierungswerkzeugen [NEI96] und ist heute als weltweiter Industriestandard für dreidimensionale Grafiken anerkannt. Es stellt eine Programmierbibliothek dar, die Funktionen für die Erzeugung, Manipulation und Animation von dreidimensionalen Modellen unter C++ bereitstellt. OpenGL wurde von Silicon Graphics als Nachfolger der Graphics Library (GL) für das proprietäre Betriebssystem IRIX entwickelt und in den letzten Jahren auf eine Vielzahl anderer Betriebssysteme, zum Beispiel Solaris (Sun), Windows 95/95/NT (Microsoft) und Linux (unabhängige Entwicklergemeinschaft) portiert. War der Einsatz von dreidimensionaler Grafik bis vor wenigen Jahren lediglich auf kostspieligen, umständlich zu bedienenden Unix-Plattformen denkbar, können heute bereits handelsübliche Mittelklasse-PCs für Simulationen mit annähernd vergleichbarem Komplexitätsgrad verwendet werden. Die hohe Standardisierung und Portabilität von OpenGL hat bei dieser Entwicklung eine entscheidende Rolle gespielt. Durch diesen Standard wurde die Entwicklung von kostengünstigen Grafikkarten mit integriertem 3D-Subsystem ermöglicht.

Für die Simulation des Fertigungsprozesses stellt OpenGL eine hervorragende Lösung dar. OpenGL bietet konzeptionell kein eigenes Dateiformat für das Speichern

von dreidimensionalen Modellen. Daher müssen entweder eigene Formate beziehungsweise Importfilter entwickelt werden oder es muss auf Standardformate wie zum Beispiel VRML zurückgegriffen werden.

4.5 Integrationsplattform

4.5.1 Middleware für verteilte Systeme

Die angestrebte Integration der einzelnen Teilmodule des verteilten Benutzerunterstützungssystems ist in konsequenter Fortführung des Gesamtkonzeptes durch die Einführung einer Dezentralisierungsplattform zu ergänzen. Die Akquisition der zu integrierenden, dezentral organisierten Daten erfordert ein hochflexibles, an unterschiedliche Anforderungen anpassbares Kommunikationssystem [FRI97]. Zu diesem Zweck soll eine Middlewareplattform in das Unterstützungssystem integriert werden, die einen hohen Standardisierungsgrad sowie ein hohes Maß an Flexibilität auf Kommunikationsebene garantiert. In Abbildung 4-2 ist der konzeptionelle Ansatz von Middlewaresystemen dargestellt.

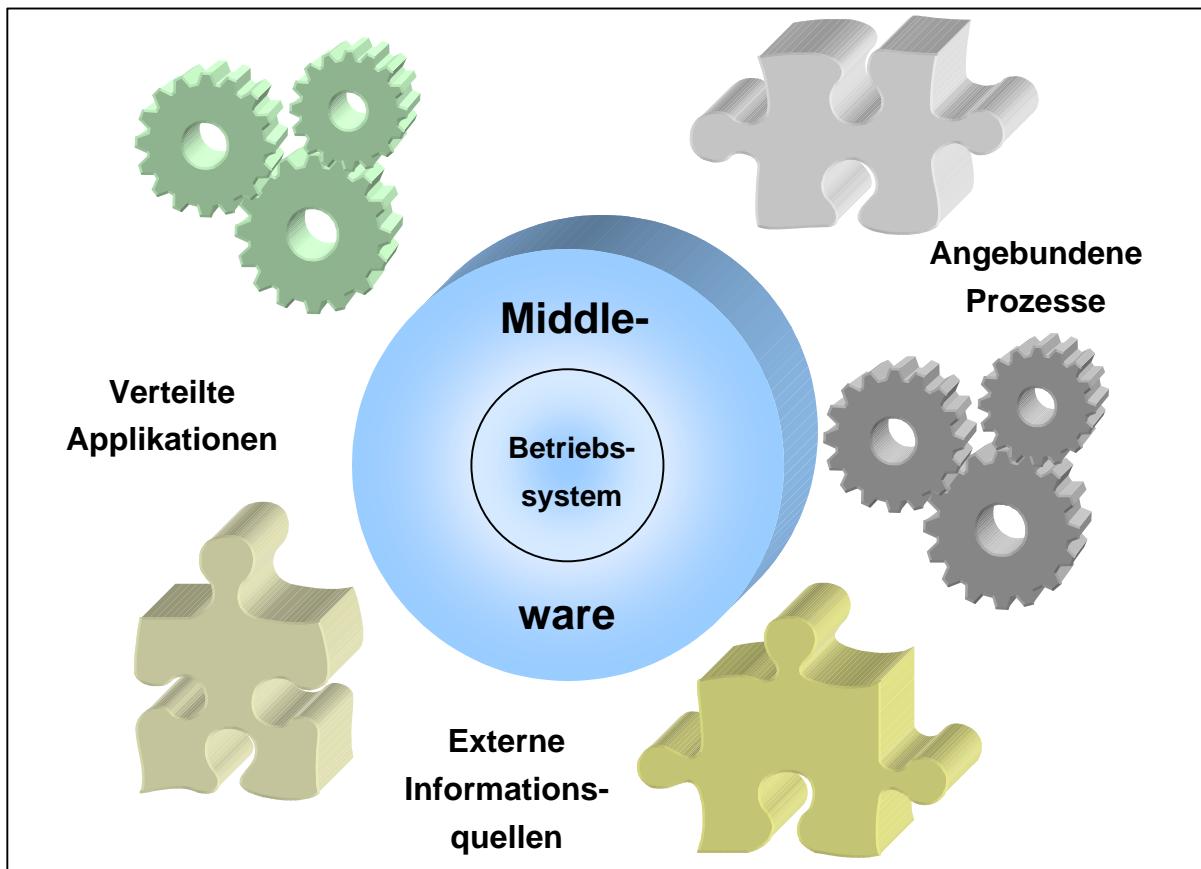


Abbildung 4-2: Einordnung von Middlewaresystemen

Middleware ist eine Menge von wenig spezialisierten Diensten, die zwischen der Systemplattform – also der Hardware und dem Betriebssystem – und den Anwendungen angesiedelt sind und deren Verteilung unterstützen.

Middlewaresysteme sind durch ihre plattformübergreifende Implementierung gekennzeichnet und setzen auf standardisierten Schnittstellen und Protokollen (zum Beispiel TCP/IP, ISO-OSI-Stack) auf. Ein solches System stellt gleichzeitig standardisierte Schnittstellen auf Applikationsebene bereit, die einen transparent verteilten Datenzugriff sowie objektspezifische Manipulationsmethoden zur Verfügung stellen [MML95].

Die folgende Übersicht fasst kurz die Vorteile von Middleware zusammen:

- ➔ Vorhandene Systeme lassen sich mit geeigneter Middleware unter Verwendung von standardisierten Interfaces an andere Systeme ankoppeln.
- ➔ Middleware beschleunigt die Entwicklungszeit neuer Kommunikationsmechanismen und -module, da sie im Vergleich zum Betriebssystem höherwertige Kommunikationsdienste zur Verfügung stellt.
- ➔ Durch die Verwendung von Middleware ist eine saubere Strukturierung der zu entwickelnden verteilten Applikationen möglich. Anwendungen können dadurch in ihrer Leistungsfähigkeit, Wartbarkeit und Erweiterbarkeit verbessert werden.
- ➔ Durch den Einsatz solcher Systeme wird die Integration von heterogenen Applikationen und Plattformen stark vereinfacht. Unterschiedliche Hardware, Betriebs- und Entwicklungssysteme können so integriert werden.
- ➔ Der hohe Standardisierungsgrad und die schnell wachsende Verbreitung stellen eine Verfügbarkeit auch für zukünftige Systeme sicher.

Die verschiedenen Middleware-Systeme lassen sich in zwei Kategorien einordnen: Die sogenannte „*general-purpose-Middleware*“, zum Beispiel OSF-DCE (Open Software Foundation - Distributed Computing Environment), CORBA (Common Object Request Broker Architecture), ANSA (Advanced Networked Systems Architecture) oder DCOM (Distributed Component Object Model) stellen Kommunikationsdienste zur Verfügung, die direkt oberhalb der Ebene des Übertragungsprotokolls aufsetzen. Im folgenden wird diese Middleware als „*generische Middleware*“ bezeichnet.

Die zweite Kategorie, die die als „*Framework-Architecture*“ bezeichnete Middleware beinhaltet, bietet neben umfassenden APIs (Application Programming Interfaces) auch integrierte Applikationen (zum Beispiel Microsoft Outlook, Lotus Notes / Domino, Novell GroupWise), die die verteilte Nutzung von Daten und Methoden dem Anwender unmittelbar ermöglichen. Diese Middleware soll im Folgenden als „*applikationsintegrierende Middleware*“ bezeichnet werden.

Im Benutzerunterstützungssystem kommen beide Middleware-Kategorien zum Einsatz.

4.5.2 Generische Middleware zur Datenintegration

Für die Integration und Manipulation von systemfremden Datenquellen ist die Implementierung einer generischen Middleware unabdingbar und stellt ein wichtiges Fundament für die Entwicklung der in Kapitel 4.4 beschriebenen aktiven Module dar.

Die Analyse der Anforderungen der aktiven Module an einen Kommunikationsstandard ergibt eine hohe Übereinstimmung mit den Spezifikationen der Middleware CORBA.

CORBA wurde unter Federführung der OMG (Object Management Group) entwickelt. Die OMG wurde 1989 von 11 Firmen gegründet, expandierte jedoch schnell und umfasst heute ca. 800 Unternehmen und Forschungseinrichtungen. Unter den Mitgliedern finden sich nahezu alle namhaften Softwarehersteller, Beratungsunternehmen und Systemintegratoren [BRA97].

CORBA ist eine Architektspezifikation für die Erzeugung, Verteilung und Verwaltung von Objekten innerhalb eines Netzwerks. Objekte sind Instanzen spezifisch deklarerter Klassen und können sowohl Variablen als auch variablenmanipulierende Funktionen, sogenannte Methoden enthalten. Die von den Methoden verarbeiteten Variablen können objektintern oder -extern vorliegen. Interne Variablen können vor Zugriff anderer Objekte geschützt werden, um Datenintegrität zu gewährleisten.

Das grundlegende Konzept hinter CORBA spiegelt sich im sogenannten ORB (Object Request Broker) wider, der im Netzwerk transparent Dienste zwischen Clients und Servern vermittelt. Jeder der beteiligten Rechner übernimmt dabei sowohl eine Client- als auch eine Serverrolle. Ein Client kann Dienste, also Daten oder Methoden vom ORB anfordern, ohne zu wissen, auf welchem Server sich dieser Dienst aktuell befindet oder von welcher Plattform dieser Dienst angeboten wird.

Für Anforderungen, die an den ORB gesendet werden, wird das sogenannte IIOP (Internet Inter-ORB Protocol) verwendet. Das IIOP basiert auf dem Internet-Protokoll

TCP/IP und kann somit in nahezu jeder Netzwerkumgebung – ob Inter- oder Intranet – eingesetzt werden [YAD96], [SCH96], [VIN97].

Die Schnittstellen zwischen Objekten werden mithilfe der IDL (Interface Definition Language) definiert. Sie stellt die Kompatibilität und den Datenaustausch der unterschiedlichen, miteinander kommunizierenden Objekte sicher.

IOP und CORBA konkurrieren mit DCOM (Distributed Component Object Model) von Microsoft. Da beide Standards bereits stark verbreitet sind und sich in bestimmten Gebieten etabliert haben, wurde von Microsoft und der OMG eine Spezifikation zur Entwicklung von Softwarebrücken zwischen diesen beiden Systemen verfasst.

CORBA wird heute entweder direkt von vielen Entwicklungssystemen unterstützt oder es existieren freie oder kommerzielle Zusatzpakete, die eine effiziente Entwicklung von CORBA-konformen Systemen ermöglichen. CORBA-Implementierungen sind heute für alle gängigen Programmiersprachen verfügbar.

Für zeitkritische Speziallösungen werden CORBA-Umgebungen bereits für einige Echtzeitplattformen, unter anderem für das in dieser Arbeit eingesetzte VxWorks, angeboten [YAS98].

4.5.2.1 Applikationsintegrierende Middleware - Framework Architecture

Neben der oben beschriebenen Anwendung generischer Middleware für die Datenintegration bedarf das Benutzerunterstützungssystem einer weiteren Integrationsplattform, die die modularisierten Teilsysteme zu einem Gesamtsystem vereint.

Es wurde bereits darauf hingewiesen, dass das Basissystem grundsätzlich auf Intranet-Technologie aufbauen sollte. Diese Basis soll nun durch den Einsatz applikationsintegrierender Middleware erweitert werden, um eine einheitliche Plattform für komplexere, oberhalb der generischen Middleware einzuordnende Dienste zu schaffen.

Bei der Auswahl dieser Plattform wurden zunächst Groupwaresysteme untersucht, die auf proprietären Protokollen bzw. Diensten aufbauen. Beispiele für solche Systeme sind Novell GroupWise oder Microsoft Outlook. Schnell stellte sich jedoch heraus, dass der augenscheinlich hohe Integritätsgrad dieser Middlewaresysteme ein Hindernis für die Ausbildung einer offenen Systemplattform für das Benutzerunterstützungssystem darstellt.

Es lassen sich zwar anhand integrierter Entwicklungsumgebungen eigene Funktionen implementieren, jedoch ist man beispielsweise bei Microsoft Outlook stark an systemeigene Funktionalitäten gebunden. Ein weiteres Argument gegen diese Plattformen ist die Geschlossenheit des Kommunikationssystems, die keinerlei Integration von systemfremden Datenquellen erlaubt.

Ein konzeptionell anders orientiertes Groupware- und Dokumentenmanagementsystem – Lotus Domino – zeichnet sich dagegen durch seine offene Architektur aus, die durch den Einsatz standardisierter Schnittstellen auf Protokoll- und Applikationsebene unterstrichen wird. Lotus Domino bezeichnet einen dynamischen Intranet- bzw. Internetserver, der relativ offen ist und durch eigene Module erweitert werden kann. Der Domino-Server bietet vielfältige Möglichkeiten der Integration weiterer Applikationen. Er kann über folgende Netzwerkprotokolle in bestehende Infrastrukturen eingebunden werden:

- ➔ IPX (Internetwork Packet Exchange)
- ➔ SPX / SAP (Sequenced Packet Exchange)
- ➔ TCP / IP (Transmission Control Protocol / Internet Protocol)
- ➔ NetBEUI (Network Basic I/O System Extended User Interface)
- ➔ BanyanVines

Weitere integrierte Netzwerk- und Kommunikationsdienste sind:

- ➔ E-Mail-Dienste: SMTP (Simple Mail Transfer Protocol), POP3 (Post Office Protocol 3), IMAP (Internet Message Access Protocol), Notes
- ➔ Gruppendiskussionen: NNTP (Network News Transfer Protocol)

Lotus Domino verfügt über eine Vielzahl von Möglichkeiten und erlaubt mithilfe standardisierter Schnittstellen die Anbindung verschiedener externer Datenquellen. Es können Datenbanken über die folgenden Dienste direkt ausgelesen bzw. manipuliert werden:

- ➔ Notes-Formeln: Lesen aus ODBC-Datenbanken (Open Database Connectivity)
- ➔ LotusScript-DataObject: Lesen und schreiben in ODBC-Datenbanken
- ➔ Oracle LSX (LotusScript) : Zugriff auf Oracle API
- ➔ Notes SQL: Externe Systeme können über SQL (Structured Query Language) auf Domino-Datenbanken zugreifen
- ➔ NotesPump: Replikation zwischen Domino, Oracle 7 und Sybase 10

Besonders wichtig ist hier die SQL- und ODBC-Kompatibilität. Über diese Standards kann der Server über das Netzwerk auf Datenbanken im Unternehmen zugreifen. ODBC ist ein offenes Programmierinterface für den Zugriff auf Datenbanken im Netz. Es unterstützt so bekannte Formate wie Access, dBase und DB2. ODBC wird als

formatabhängiger Dienst auf der Seite der Datenbank als Dienst aktiviert, sodass der externe Zugriff mit entsprechender Authentifizierung erfolgen kann.

ODBC wird meist in Kombination mit SQL verwendet. SQL abstrahiert Datenbankzugriffe so, dass sie formatunabhängig implementiert werden können. ODBC bildet den formatunabhängigen Zugriff auf das spezifische Datenformat ab. SQL ist anerkannter ANSI- und ISO-Standard.

Ein weiterer Vorteil des Lotus-Servers ist die Möglichkeit der Integration anderer Applikationen über deren Dateiformate. Als Beispiel sei eine Tabelle genannt, deren Inhalt von einer Datenbank mehrmals täglich aktualisiert wird. Diese Tabelle kann Domino als Datei einbinden und auf Anfrage eines Clients „online“ in den Internet-Standard HTML konvertieren. Bei jedem Aufruf der Tabelle erhält der Client somit die jeweils aktuellen Daten.

Lotus Domino ist in der Lage, alle Dateiformate, die OLE (Object Linking and Embedding) unterstützen (zum Beispiel MS Office, Micrografx Suite, AutoCAD 2000) online zu integrieren.

Applikationen für den dynamischen Webserver können auf verschiedene Weise programmiert werden. Als systemeigene Entwicklungsumgebung dient der Lotus Designer, der die Entwicklung eigener Masken, Menüs und Layouts sowie die Programmierung mit LotusScript vorsieht. Zusätzlich ist die Verwendung anderer Programmiersprachen bzw. Entwicklungssysteme wie JavaScript, -CGI (perl, C, C++), ActiveX und JAVA möglich.

Lotus Domino ist historisch aus dem im industriellen Umfeld als De-facto-Standard geltenden Messaging- und Dokumentenmanagementsystem Lotus Notes entstanden und erweitert die Funktionen des Notes-Servers um viele wichtige intranetorientierte Dienste.

Domino-Server unterstützen Dienste zur Verteilung der Datenbanken sowie die vollautomatische Replikation, das heißt den Datenabgleich mehrerer Server an unterschiedlichen Standorten, entweder nach einem bestimmten Zeitschema oder durch eine automatische Aktualisierung bei der Änderung eines Datenbestandes. Mithilfe dieser Technik ist es möglich, das Produktionsunterstützungssystem zwischen verschiedenen Werkstätten oder verschiedenen Niederlassungen eines Unternehmens abzulegen.

Die Replikationsfunktion erhöht Datensicherheit und die Verfügbarkeit des Systems, da ein ausgefallener Server jederzeit ersetzt und die Datenbanken vom Replikat zurückkopiert werden können. Damit kommt Domino den Forderungen nach Systemstabilität und Sicherheit nach.

Eine weitere fundamentale Eigenschaft des Servers ist die Möglichkeit der Vergabe von Benutzerrechten. Obwohl das System im wesentlichen Internetdienste für die Client-Server-Kommunikation nutzt, wurde das Zugriffskonzept von Domino konsequent umgesetzt. So ist es möglich, dass Seiten, die im Web-Browser betrachtet werden, in Abhängigkeit vom Anwender unterschiedlichen Inhalt haben bzw. nicht sichtbar sind.

5 Systemarchitektur

Das verteilte prozessunterstützende System für die Fertigung in industriellen Umgebungen basiert auf einem generischen Konzept, das einerseits spezifisch genug für konkrete Applikationen ist, andererseits aber auch genügend Freiraum für zukünftige Adaptionen des Systems an weitere Umgebungen erlaubt. Das Konzept wird in einer Softwarearchitektur umgesetzt, die als Integrationsplattform für die verschiedenen Technologiemodule dient.

5.1 Anwendungsfälle

Grundsätzlich soll das Unterstützungssystem als einheitliche, eine Vielzahl von Diensten integrierende Plattform fungieren und damit im Gegensatz zu existierenden, nicht anwenderorientierten Insellösungen stehen, die über keine einheitliche Bedienoberfläche verfügen und nur bestimmte Anwendungsfälle abdecken. Das System soll Eigenverantwortlichkeit, Entscheidungsfreiheit und damit die Motivation der Werker steigern.

In Anlehnung an die aus dem holonischen Fertigungssystem folgenden Forderungen muss das Unterstützungssystem durch entsprechende Kommunikations- und Informationssysteme flexible Teamstrukturen in den Werkstätten fördern.

5.1.1 Information

Ein wesentlicher Aspekt ist die Integration, Aufbereitung und Präsentation von Informationen aus verschiedenen Bereichen des Arbeitsumfelds [WAA96], [MEC95]. Mit der zusätzlichen Information ist eine Aufwertung von weniger erfahrenen Werkern und Auszubildenden verbunden, da diese über das System Zugang zu Expertenwissen erhalten. Im System verfügbare Informationen erstrecken sich auf folgende Bereiche:

- ➔ Arbeitsanleitungen: Bedienungsanleitungen von Maschinen, Anleitung zur Ausführung von Prozessen, Informationen zum Arbeitsablauf innerhalb und außerhalb der Werkstatt

- ➔ Integration des Unternehmens-Intranets: Abruf unternehmensweit verfügbarer Informationen, z.B. Nutzung des Telefonverzeichnisses oder des Terminkalenders
- ➔ Sicherheitsinformationen: Richtlinien und Spezifikationen, Unfallstatistiken, häufige Unfallursachen, Ansprechpartner
- ➔ Fertigungs- und Qualitätsstandards: Dokumentation der anzuwendenden Fertigungsstandards und der einzuhaltenen Toleranzen, Informationen zur geforderten Qualität
- ➔ Wartungsunterlagen und -pläne: Terminpläne für die Wartung von Maschinen und Werkzeugen, Dokumentation ausgeführter Wartungsarbeiten, Anleitung zur Ausführung von Wartungsarbeiten

Die Informationen müssen entsprechend der bereits erläuterten Forderungen strukturiert dargestellt werden.

5.1.2 Werkerkommunikation

Im Sinne der Förderung der Arbeit in Teams und der Verbesserung des Informationsaustausches innerhalb der Teams, innerhalb der Werkstatt und zwischen den verschiedenen Abteilungen des Unternehmens müssen neue Kommunikationsmechanismen eingeführt werden. Die Mechanismen sollten so gestaltet sein, dass sie dazu beitragen, orts- und zeitbedingten Informationsverlust zu vermeiden. Dieser Informationsverlust entsteht beispielsweise dadurch, dass die Werker im Regelfall keine Möglichkeit haben, konstruktionsbedingte Fertigungsprobleme den Konstrukteuren selbst mitzuteilen oder dadurch, dass zwischen den Schichten der Werkstatt kein Austausch stattfindet.

Die Modi der in Kapitel 3.2 definierten horizontalen und vertikalen Kommunikation lassen sich wie folgt genauer spezifizieren:

➔ Horizontale Kommunikation:

- Innerhalb eines Teams: Austausch von Erfahrungen, Expertenwissen, Stellen von Fragen, Organisatorisches
- Zwischen Teams: Ähnlicher Informationsaustausch wie innerhalb eines Teams, Teams können so lokales Expertenwissen global zur Verfügung stellen
- Entlang eines Teilproduktflusses durch die Werkstatt: Werkstück- bzw. auftragspezifische Informationen begleiten das Teilprodukt entlang der verschiedenen Stationen

- Zwischen Teilproduktflüssen: An bestimmten Stationen werden Informationen ausgetauscht, die diesen Prozess an verschiedenen Werkstücken betreffen

➔ **Vertikale Kommunikation:**

- Zwischen Werkern und anderen Abteilungen des Unternehmens: Austausch von Informationen mit Planung, Konstruktion, Auftragsmanagement etc.
- Zwischen Werkern und dem Unternehmen als Ganzem: Austausch von Informationen über das unternehmensweite Intranet, z.B. Abruf von Telefonnummern oder Terminen

Grundsätzlich sind alle Kommunikationswege interaktiv zu gestalten, das heißt zum Beispiel, dass eine Nachricht zitiert und beantwortet werden kann, Nachrichtenlisten nach Themen, Datum und Autor sortiert und nach Stichworten durchsucht werden können. Des weiteren sollten Themengruppen eingeführt werden, um eine grobe Kategorisierung bereits vorab zu treffen.

5.1.3 Prozessintegration und Verifikation

Die Flexibilisierung des Fertigungsablaufs durch die Förderung von Teamarbeit erfordert die Unterstützung von weniger erfahrenen Werkern. Entsprechende Funktionen müssen daher zur direkten Prozessunterstützung integriert werden.

Als Datenbasis für die Integration der Prozessunterstützung muss der reale Prozess dienen. Die hieraus gewonnenen Prozessinformationen wie z.B. Parameter oder Informationen über einen kinematischen Vorgang müssen auf geeignete Weise aufbereitet und für die Verwendung durch den Werker zugeschnitten werden. Gleichzeitig sollen dem Werker Funktionen zur Verfügung gestellt werden, die eine automatische oder teilautomatische Überwachung bzw. einfache Regelung des Prozesses erlauben.

Zur Verifikation eines automatisierten Prozesses vor seiner Ausführung müssen Werkzeuge zur Simulation und Verifikation des Maschinenprogramms geschaffen werden. Während der Prozessausführung ist eine Überwachung bzw. Regelung zur Entlastung des Werkers gefordert.

Grundsätzlich müssen die Visualisierungs- und Simulationswerkzeuge werkergerecht gestaltet werden. Die Gestaltung der Benutzeroberfläche sowie die Verwendung moderner 3D-Visualisierungsmethoden sollen dieser Anforderung Rechnung tragen.

5.1.4 Aufbereitung von Technologiedaten

Im Bereich der industriellen Fertigung werden heute meist Zeichnungen oder Parameterlisten für die Übermittlung von Fertigungsaufträgen verwendet. Beide Übermittlungsformen setzen beim Werker großes Erfahrungswissen voraus und führen immer wieder zu Problemen und Fehlern in der Fertigung. Ziel des Unterstützungssystems muss sein, Werkzeuge für die interaktive Visualisierung von Werkstücken anzubieten. Diese Werkzeuge müssen die werksinternen Produktdatenbanken lesen und deren Informationen aufbereiten können. Die Darstellung soll möglichst realistisch und interaktiv gestaltet sein. So sollen unter anderem einfach zu bedienende Zoom- und Rotationsfunktionen integriert sein.

5.1.5 Produktionsplanung

Die Produktionsplanung umfasst die Verteilung von Aufträgen auf Ressourcen sowie den zeitlichen Ablauf der Produktion. Die Produktionsplanung wird in den untersuchten Werkstätten ausschließlich durch die Vorarbeiter ausgeführt. Sie erhalten die Planung in elektronischer Form, drucken sie aus und verteilen dementsprechend die Aufträge an die einzelnen Werker. Das System verhindert aufgrund der starren Organisation und der fehlenden Interaktivität eine flexible Anpassung der Pläne an die aktuellen Randbedingungen.

Ziel des Unterstützungssystems muss die Integration einer interaktiven, für den Werker zu beeinflussenden Planungsumgebung sein. Die Planungsumgebung muss dabei so gestaltet sein, dass sie vollautomatisch Randbedingungen wie Termine und Überschneidungen von Ressourcenauslastungen berücksichtigt. Das Planungssystem muss automatisch unterschiedliche Planungsvorschläge bewerten und zwischen den Arbeitsstationen abgleichen [SEP98], [PAO99], [OHW98].

Die Einführung eines solchen Planungswerkzeugs ist essentiell für die Flexibilisierung des Produktionsablaufs [VML98].

5.1.6 Bewertung der Produktion

Das Produktionssystem Werkstatt soll durch die Entwicklung von Evaluierungswerkzeugen in die Lage versetzt werden, sich autonom zu optimieren. Die Bewertung der Einhaltung des Produktionsplanes steht hier im Vordergrund der Anforderungen. Durch den direkten Vergleich der Termine der Planung mit den realen Daten wird dem Werker die Möglichkeit gegeben, Planungsschwächen und Engpässe in der Fertigung zu lokalisieren und zu beheben. Die Termine müssen automatisch verglichen und die Planabweichungen in geeigneter Form dargestellt werden. Eine grafische Anzeige soll dabei das Auffinden von Häufungen der Planabweichung erleichtern.

5.2 Systemarchitektur

Ziel des prozessunterstützenden Systems ist die Bereitstellung einer Softwareplattform, die als Basis für anwendungsspezifisch zugeschnittene Lösungen in unterschiedlichen industriellen Fertigungsumgebungen dienen kann. Das zugrundeliegende Softwarekonzept muss daher relativ hohen Anforderungen in Bezug auf

- ➔ Flexibilität,
- ➔ Stabilität,
- ➔ Skalierbarkeit und
- ➔ Konfigurierbarkeit

genügen.

Des weiteren muss es in der Lage sein, die in den vorhergehenden Kapiteln beschriebenen inhaltlichen und funktionellen Anforderungen zu erfüllen. Die geforderte Flexibilität und Skalierbarkeit, also die Möglichkeit, das System an andere Werkstätten oder andere Branchen anzupassen zu können, lässt sich nur mit einem modular aufgebauten System erreichen. Einzelne Module können somit ausgetauscht, hinzugefügt oder entfernt werden, ohne dass das Grundkonzept geändert werden muss.

Die Einzelmodule müssen mit Hilfe standardisierter Kommunikationsmechanismen in die Lage versetzt werden, Daten schnell und effizient auszutauschen. Das System muss trotz seiner Modularität nach außen kompakt und homogen erscheinen. Aus diesem Grund sind sogenannte Styleguides, also Richtlinien für ein homogenes Moduldesign im Hinblick auf Kompatibilität, Kommunikation und Layout zu definieren. Die einzelnen Module sollen einer integrativ und konfigurativ arbeitenden Systemkomponente untergeordnet werden.

Die Modularisierung des Unterstützungssystems wird durch die Anwendung von internationalen Standards insbesondere für die Kommunikation stark vereinfacht und die spätere Erweiterbarkeit um neue Module so gewährleistet.

Die in Abbildung 5-1 gezeigte Integrationskomponente steht exemplarisch für die auf der jeweiligen Systemebene implementierte Betriebssoftware, die den Rahmen um

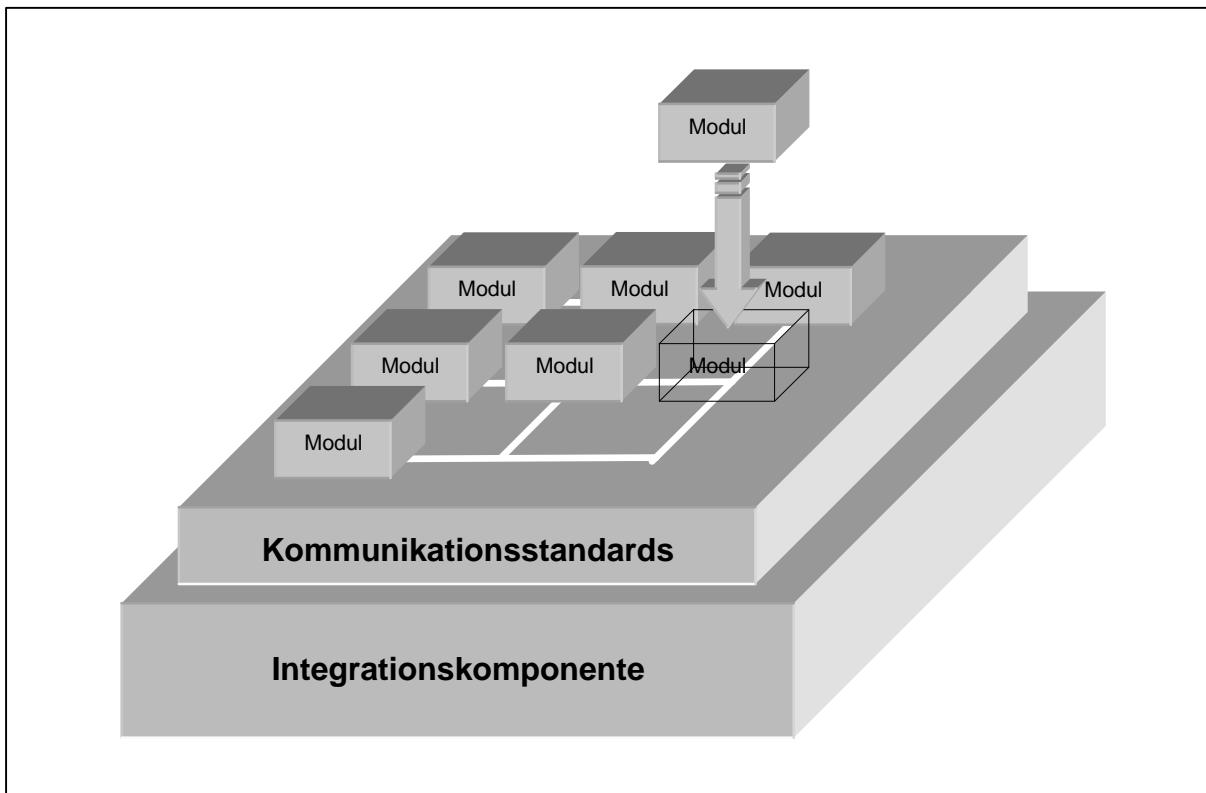


Abbildung 5-1: Modulare Architektur

die einzelnen Module stellt und die Kommunikation zwischen den Modulen dieser und anderer Ebenen koordiniert.

5.2.1 Funktionsorientiertes Ebenenmodell

Die verschiedenen Aufgaben des Unterstützungssystems lassen sich in drei funktionale Ebenen unterteilen (Abbildung 5-2):

- ➔ Akquisition von Informationen,
- ➔ Integration der Daten sowie die
- ➔ Präsentation der Inhalte und Systemfunktionen.

Die Funktionen der Akquisitionsebene dienen der datentechnischen Anbindung von Fertigungsprozessen und externen Datenbanken an das Unterstützungssystem. Diese Daten werden durch die Funktionen der Integrationsebene aufbereitet und mit weiteren Informationen wie z.B. den aktuellen Benutzerdaten verknüpft, um schließlich mithilfe der Funktionen der Präsentationsebene interaktiv dargestellt zu werden.

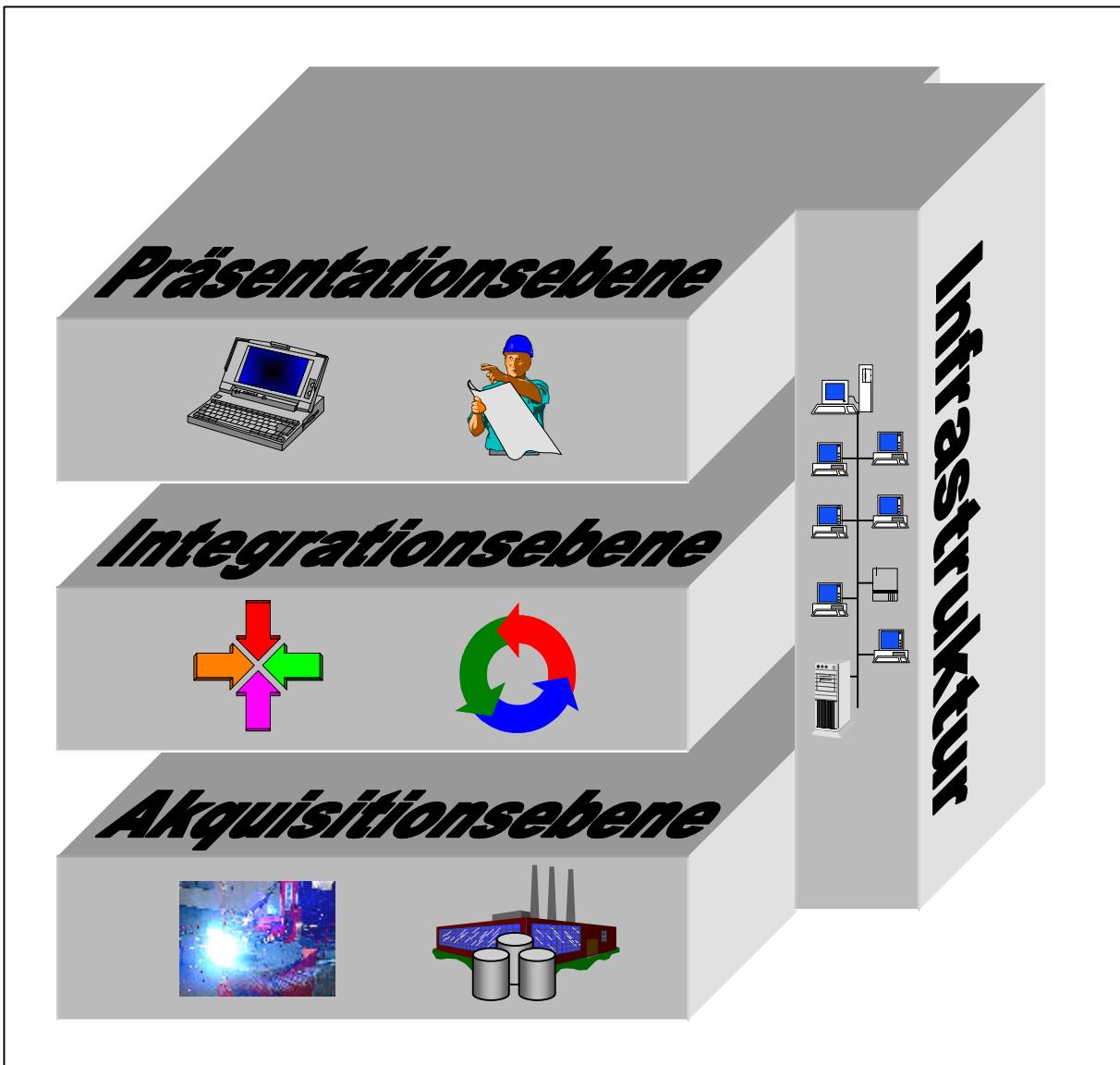


Abbildung 5-2: Funktionsorientiertes Ebenenmodell

Die einzelnen Ebenen sind weitestgehend unabhängig voneinander implementiert und kommunizieren lediglich über die gemeinsam genutzte Infrastruktur miteinander. Diese Architektur bietet ein sehr hohes Maß an Stabilität und Sicherheit, da zeitkritische Prozesse getrennt von an die Interaktion von Mensch und Maschine gebundenen Prozessen ablaufen können. Ein zusätzlicher Vorteil ist die in der Architektur verankerte Abschirmung des Benutzers von Systemprozessen und Datenbanken und die damit verbundene Sicherheit dieser Module.

In den folgenden Abschnitten werden die Aufgaben der einzelnen Ebenen genauer erläutert.

5.2.1.1 Akquisitionsebene

Das Prozessunterstützungssystem soll den Benutzer nicht vom Prozess distanzieren, sondern ihm neue Möglichkeiten zur Überwachung und zum Vergleich von Prozessen sowie zur Optimierung der Fertigung anbieten.

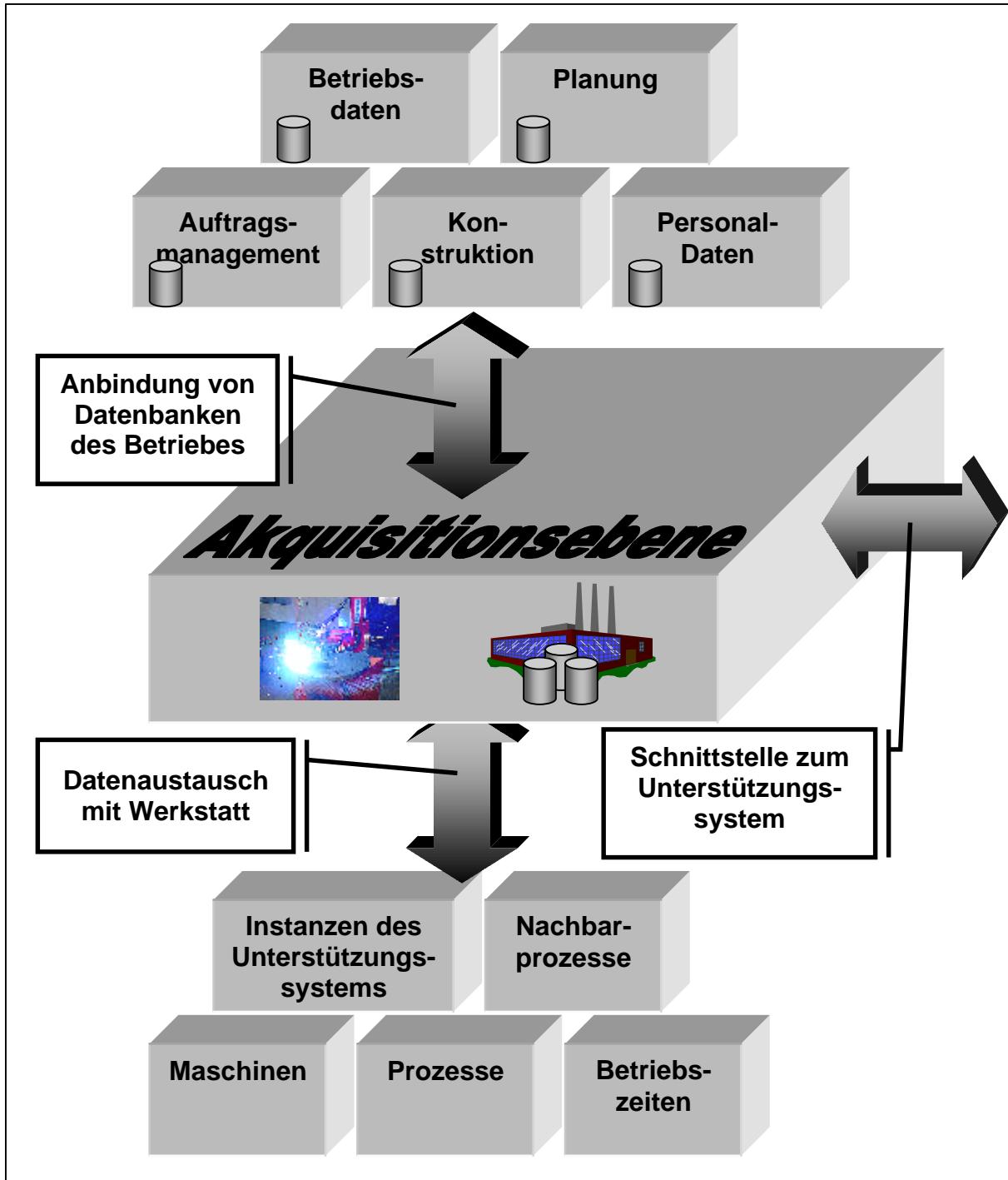


Abbildung 5-3: Architekturkomponente Akquisitionsebene

Es muss das Ziel eines solchen Systems sein, basierend auf aktuellen Daten aus dem Fertigungsprozess und weiteren Informationsquellen inner- und außerhalb der Werkstatt Entscheidungshilfen zur Verfügung zu stellen [MEG93].

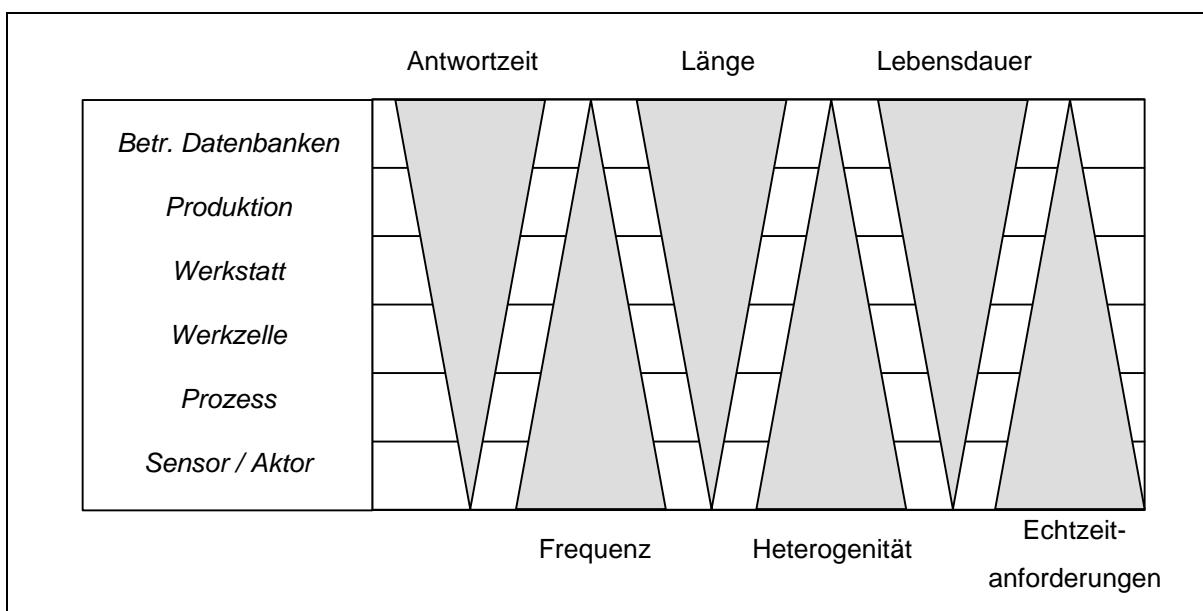
Da die Anbindung des Systems an externe Datenquellen Risiken in Bezug auf

- ➔ die Stabilität und das Zeitverhalten der angekoppelten Systeme,
- ➔ die Stabilität des Unterstützungssystems selbst und
- ➔ die Sicherheit der angekoppelten Systeme

birgt, wurden die betreffenden Kommunikationsfunktionen in einer separaten Ebene gekapselt. Die Funktionen stellen, wie in Abbildung 5-3 gezeigt wird, die Verbindung zu verschiedenen Datenquellen auf Betriebs- und Werkstattebene zur Verfügung und vermitteln die gewonnenen Daten weiter an die Integrationsebene.

Dabei müssen unterschiedliche Anforderungen der Schnittstelle zu betrieblichen Datenbanken und der Schnittstelle zur Werkstatt beachtet werden. In Anlehnung an das von [GAL95] beschriebene Modell der Kommunikationsanforderungen einer automatisierten Arbeitszelle lässt sich das Anforderungsspektrum des Unterstützungssystems, wie in Abbildung 5-4 gezeigt, spezifizieren.

Man erkennt, dass die Nachrichtenfrequenz, ihre Heterogenität sowie die Echtzeitanforderungen der Kommunikation mit der Nähe zum Fertigungsprozess zunehmen. Eine kurze Antwortzeit und Lebensdauer der kurzen Nachrichten sind hier notwendig. So muss die Anbindung von Sensoren und Aktoren die höchsten Anforderungen an das Echtzeitverhalten erfüllen. In den darüber angeordneten Ebenen nehmen diese Forderungen kontinuierlich ab.



*Abbildung 5-4: Kommunikationsanforderungen der automatisierten Fertigung
[GAL95]*

Im Bereich der werkstattweiten Kommunikation sowie auch in der Kommunikation der Werkstätten untereinander (Produktion) überwiegen die Forderungen nach Nachrichtenlänge und -lebensdauer, während Echtzeitanforderungen nur eine

untergeordnete Rolle spielen und Antwortzeiten üblicherweise deutlich länger ausfallen. Betriebliche Datenbanken schließlich erfordern praktisch keine echtzeitfähige Kommunikation. Sie transferieren und speichern große Datenvolumina bei relativ langen Antwortzeiten. Für das Unterstützungssystem sind im wesentlichen die in Tabelle 5-1 abgebildeten betrieblichen Datenbanken relevant.

Bereich	Daten	Beispiel
Konstruktion	CAD-Zeichnungen	Übersichtsplan des Schiffssegmentes
Planung	Planungsinformationen in verschiedenen Hierarchie- und Detailstufen	Arbeitsplan für die Montagewerkstatt
Betriebsdaten	Arbeitskosten, Materialwirtschaft	Statusabfrage des Materiallagers
Auftragsmanagement	Werkstattaufträge	Einzelauftrag Rohrschweißen
Personal	Informationen zum Personal	Urlaubsplanung

Tabelle 5-1: Relevante betriebliche Datenbanken

Die technische Umsetzung der betrieblichen Datenintegration muss über eine standardisierte Datenbankschnittstelle realisiert werden, damit das System für weitere Integrationen offen ist. In kommerziellen Datenbanken relativ weit verbreitete Technologien sind ODBC („Open Database Connectivity“) und SQL („Structured Query Language“), die beide als De-facto-Standard in industriellen Datenbankanwendungen bezeichnet werden können. Mit diesen Standards ist ein netzwerkweiter Zugriff auf bestehende Datenbanken zu realisieren.

Im Bereich der werkstattbezogenen Datenintegration bestehen nach dem Kommunikationsmodell andere Anforderungen an die Kommunikationsmechanismen. Insbesondere die Anbindung von zeitlich variablen Werten, wie z.B. Maschinen- bzw. Prozessparametern, erfordert eine hohe Verarbeitungs- und Übermittlungsgeschwindigkeit der gewonnenen Daten bei gleichzeitiger Entkopplung von Fertigungsprozess und Unterstützungssystem. Diese Entkopplung ist notwendig, um die Kontinuität und damit die Stabilität der Prozesse nicht zu gefährden [YAS98].

Die beschriebenen Anforderungen lassen sich nur mit einem echtzeitfähigen Akquisitionsmodul realisieren, das in der Lage ist, schnell Daten aus dem Prozess zu sammeln, zu filtern und mit Hilfe eines integrierten, zeit-unkritischen Anbindungsmoduls ins Netzwerk zu verteilen (Abbildung 5-5).

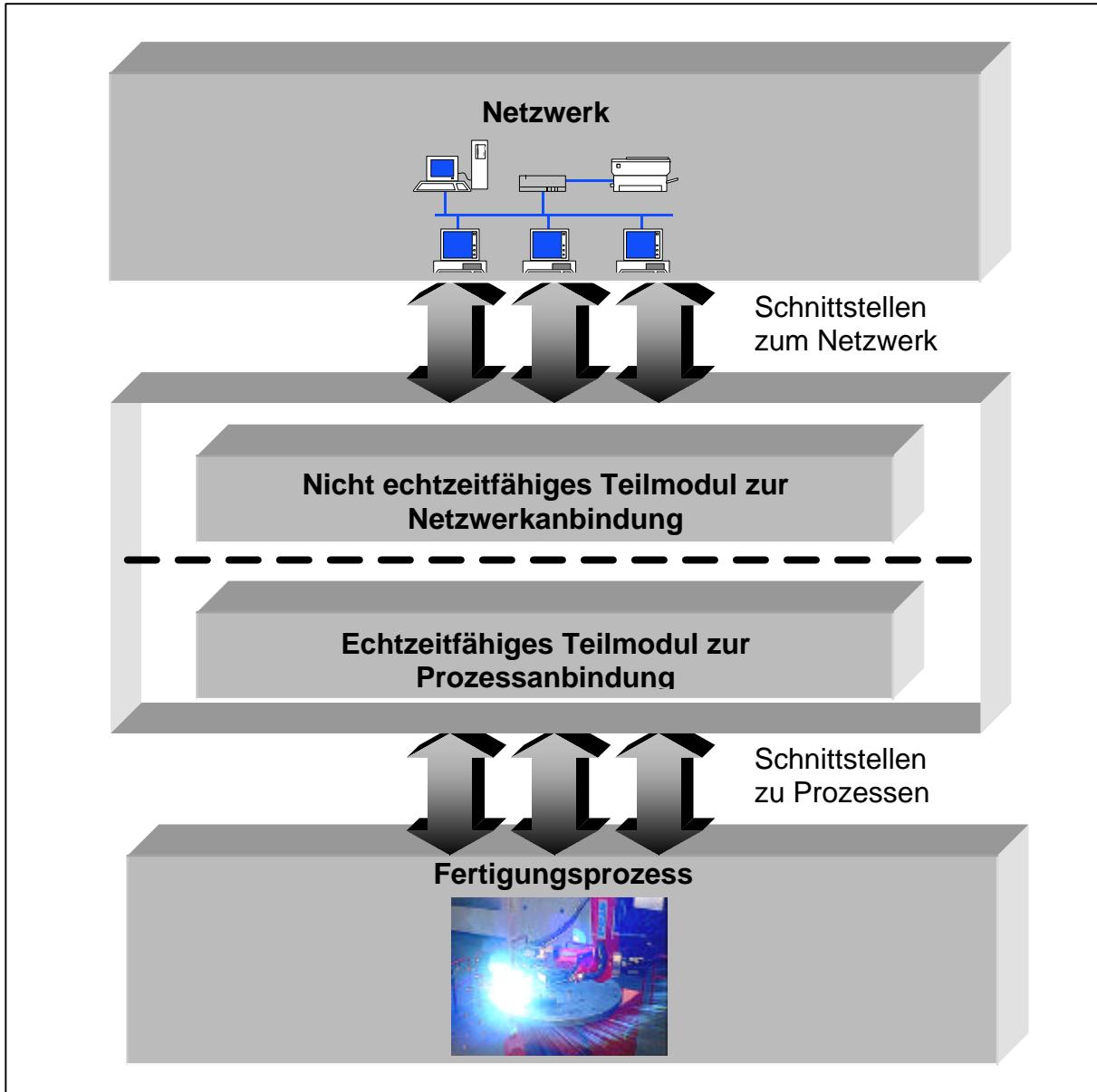


Abbildung 5-5: Echtzeitfähiges Akquisitionsmodul

Die Kommunikation zwischen zeitkritischen und nicht zeitkritischen Funktionen der Akquisitionsebene macht Datenpuffer (sog. „queues“) erforderlich, die Datensätze zwischenspeichern können, um sie auf Anfrage an die nicht zeitkritischen Funktionen weiterzuleiten. In Abhängigkeit von spezifischen Kontinuitäts- und Aktualitätserfordernissen der jeweiligen Datenquelle ist abzuwegen, ob der verwendete Puffer als „first-in-first-out“ (fifo) oder „last-in-first-out“ (lifo) anzulegen ist. Die Schnittstelle zum Fertigungsprozess muss über ein geeignetes Bussystem erfolgen. Der Bustyp wird oft durch bereits bestehende Schnittstellen der Maschinen vorgegeben und muss vom echtzeitfähigen Teilmodul unterstützt werden. In der Fertigungstechnik weit verbreitete Bussysteme sind z.B. PROFIBUS [DIN19], FIP [SOL93] und CAN [ISO93]. Im Bereich offener Steuerungen hat sich die hardwareseitige Spezifikation des VMEbus [VIT95], [OBS95] durchgesetzt.

Bei Daten, die für eine spätere Prozesssimulation verwendet werden sollen, ist eine lückenlose und kontinuierliche Aufzeichnung erforderlich. In diesem Fall bietet sich die Verwendung eines möglichst groß dimensionierten fifo-Puffers an. Steht bei bestimmten Daten, z.B. bei Prozessparametern, die überwacht oder online visualisiert werden sollen, die Aktualität im Vordergrund, so ist ein lifo-Puffer zu implementieren.

Je nach Anwendungsfall sind Sicherheitsmechanismen vorzusehen, die systeminterne Warn- oder Fehlermeldungen ausgeben, falls ein Überlauf eines kritischen Puffers auftritt. Zusätzlich ist der Benutzer über die Aktualität von Daten in Form von Zeitangaben oder farblichen Kennzeichnungen zu informieren.

Des weiteren muss die Akquisitionsebene standardisierte Schnittstellen zur Integrationsebene besitzen, die die Informationen schnell und möglichst transparent transportieren. In diesem Bereich steht nicht die Echtzeitfähigkeit der Schnittstellen im Vordergrund, sondern

- ➔ die Plattformunabhängigkeit,
- ➔ ihr Implementierungsaufwand und
- ➔ die Portabilität.

So kann sichergestellt werden, dass die Erweiterbarkeit und Übertragbarkeit des Benutzerunterstützungssystems gegeben ist. Die verwendeten Protokolle müssen zu vorhandenen Infrastrukturen kompatibel sein, sodass sich die Anwendung der TCP/IP-Protokollfamilie anbietet.

Auf höherer Protokollebene sind die Middlewarestandards CORBA („Common Object Request Broker Architecture“) und OSACA („Open System Architecture for Controls within Automation Systems“) in Betracht zu ziehen. Durch ihre Verwendung kann das Benutzerunterstützungssystem einer Vielzahl von Datenbankapplikationen und Maschinensteuerungen geöffnet werden.

5.2.1.2 Integrationsebene

Die Verbindung zwischen Akquisitionsebene und Präsentationsebene wird durch die Integrationsebene hergestellt. Ihre Aufgabe ist das Zusammenstellen, Aufbereiten und Verknüpfen von Rohdaten aus dem Prozess und aus den betrieblichen Datenbanken. Die Integrationsebene bereitet mit Hilfe entsprechender Funktionen und Algorithmen die Rohdaten für die Darstellung durch die Präsentationsebene vor.

Standardisierte Schnittstellen stellen auch hier die Kompatibilität und spätere Erweiterbarkeit sicher.

5.2.1.3 Präsentationsebene

Die Präsentationsebene soll die in der Akquisitionsebene gewonnenen und in der Integrationsebene aufbereiteten Daten in geeigneter Form darstellen.

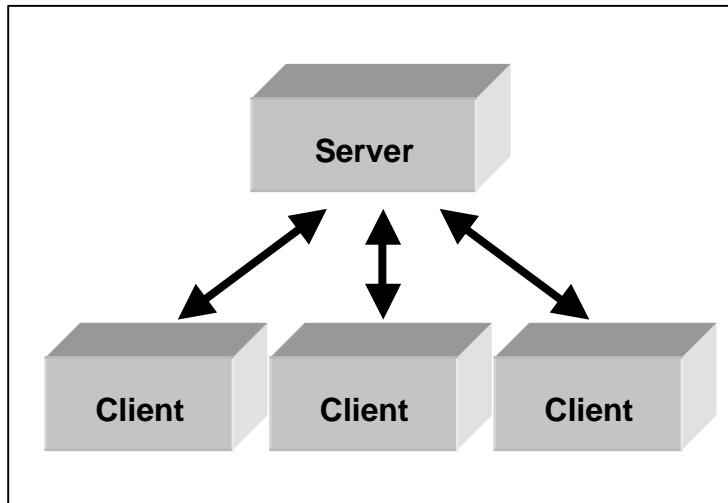


Abbildung 5-6: Client-Server-Architektur

Die Präsentationsebene stellt die Schnittstelle zwischen dem Benutzer und dem Unterstützungssystem dar. Sie muss „vor Ort“ einsatzfähig, das heißt, auf den in Werkstätten eingesetzten PCs lauffähig sein. Da das System multi-userfähig sein muss, müssen beliebig viele Instanzen der Funktionen der Präsentationsebene parallel laufen können. Das Subsystem Integration/Präsentation stellt eine Client-Server-Architektur (Abbildung 5-6) dar. In diesem Zusammenhang ist besonders auf die ausreichende Dimensionierung des Servers Wert zu legen.

5.2.2 Teilmodule

Die oben beschriebene Ebenenarchitektur gestattet die weitere Granulierung der innerhalb der Ebenen agierenden Softwaremodule. Die Granulierung stützt sich im wesentlichen auf die in Kapitel 5.1 identifizierten Anwendungsfälle.

5.2.2.1 Benutzeroberfläche

Die Benutzeroberfläche hat die Aufgabe, Informationen den Anforderungen an Ergonomie und Informationsgehalt entsprechend zu vermitteln. Sie stellt die Kommunikationsschnittstelle zwischen Anwender und Unterstützungssystem dar.

Die Anpassung an die individuellen Kenntnisse und Qualifikationen jedes Benutzers macht die Entwicklung einer adaptiven, sich automatisch den Benutzerspezifikationen anpassenden Oberfläche notwendig [NOE98]. Gleichzeitig muss sichergestellt werden, dass die Client-Seite den Forderungen nach

Kompatibilität und Werkstatt-Tauglichkeit genügt [HMS99]. Die Client-Software muss also auf gängigen PC-Systemen einsetzbar sein, ohne dass dadurch Nachteile in Bezug auf die Adaptivität der Oberfläche entstehen.

Wichtiges Merkmal bei der Konzeptionierung des Systems ist die Trennung der funktionalen Benutzeroberfläche von Inhalten sowie die Möglichkeit, Oberfläche und Inhalte getrennt voneinander administrieren zu können. Die Inhalte sollen mit Hilfe von separat abgelegten Layout- und Benutzerdaten werkstattgerecht und benutzerbezogen dargestellt werden (Abbildung 5-7).

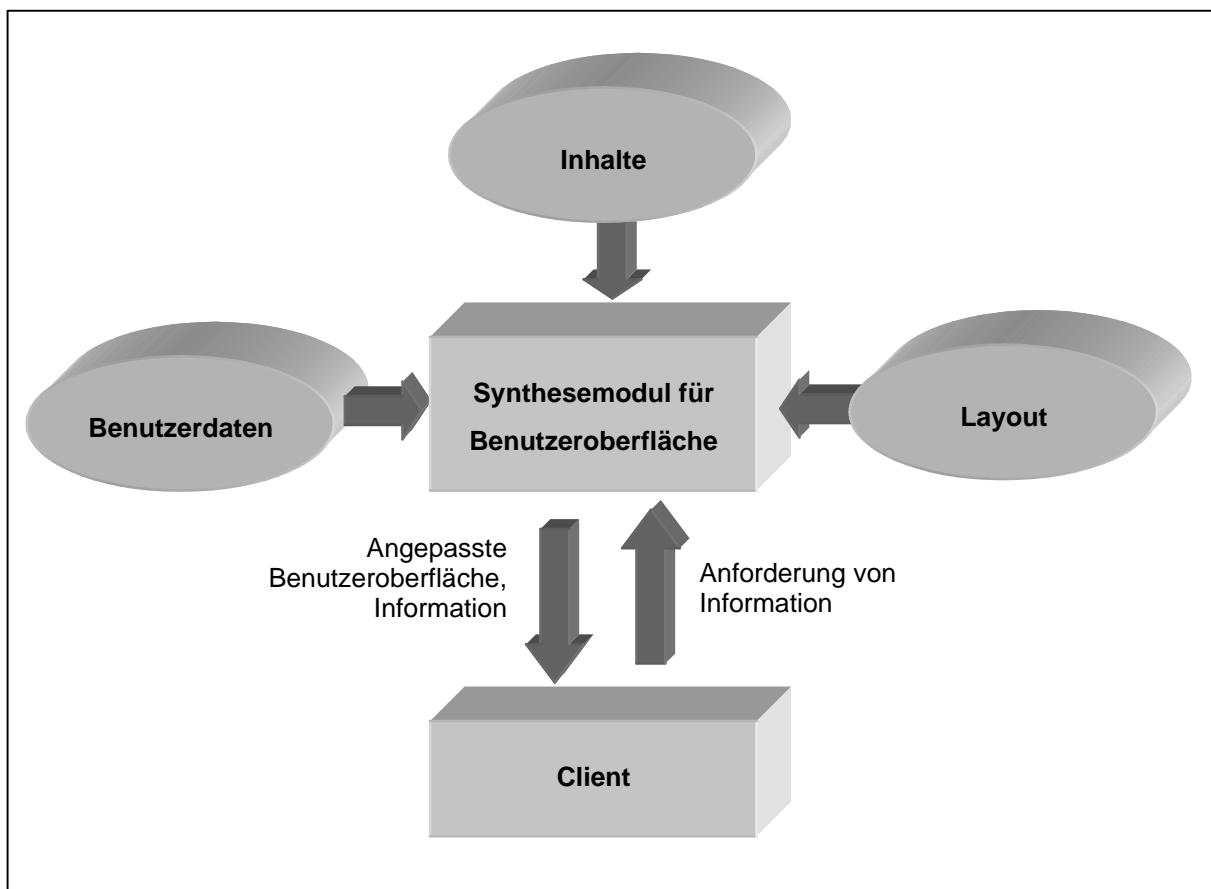


Abbildung 5-7: Synthese der Benutzeroberfläche

Diese Technik wird heute bereits im Bereich von Intra- und Internetservern angewendet.

5.2.2.2 Visualisierung prozessrelevanter Daten

Die Definition eines Fertigungsprozesses erfolgt mithilfe der Spezifikation der technologischen Parameter und der genauen Definition des Endproduktes. Die Definition des Endproduktes kann entweder – wie in der Montagewerkstatt für

Strukturelemente – über eine technische Zeichnung oder – wie in der Rohrwerkstatt – über eine Liste von Parametern bzw. textuelle Skizzierung erfolgen.

Das Unterstützungssystem soll vorhandene Produktmodelle, die dem Werker derzeit nicht zur Verfügung stehen, aufbereiten und interaktiv darstellen. Die Produktmodelle basieren im wesentlichen auf CAD-Daten, die vom System ausgelesen, zur Benutzeroberfläche übertragen und dort visualisiert werden müssen.

Der Server des Unterstützungssystems generiert bei Bedarf Abfragen der Produktmodelldatenbanken, die in der Konstruktionsabteilung vorliegen. Das zum Server übertragene Produktmodell wird anschließend vom Server an das Visualisierungsmodul des Clients weitergereicht und dort in einer schattierten, frei dreh- und zoombaren 3D-Grafik dargestellt (Abbildung 5-8).

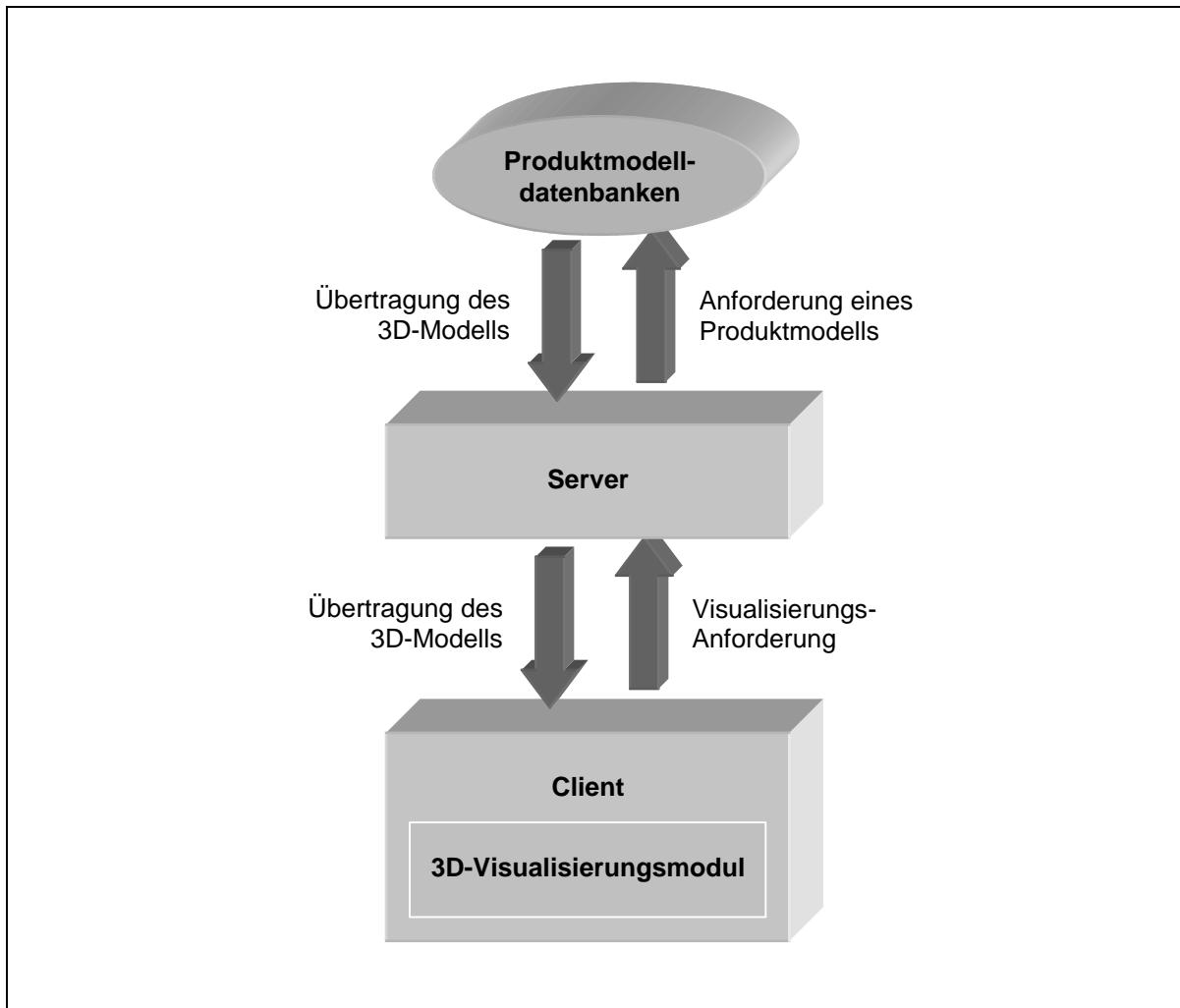


Abbildung 5-8: Architektur des Visualisierungsmoduls

5.2.2.3 Echtzeitplattform und exemplarische Werkzelle

Das Prozessunterstützungssystem kommuniziert direkt mit der Fertigungsumgebung. Dabei stellt es dem Anwender möglichst zeitnah Daten aus den Prozessen zur Verfügung und schafft Interaktionsoptionen.

Im Zuge dieser Arbeit ist eine Echtzeitplattform in eine exemplarische, für industrielle Anwendungen typisch konfigurierte Arbeitszelle integriert worden. Die Integration dieser Arbeitszelle ermöglicht es, Testläufe innerhalb einer realistischen Fertigungsumgebung durchzuführen und das System in Bezug auf seine Leistungs- und Integrationsfähigkeit zu optimieren.

Ziel der Integration der Echtzeitplattform in das Gesamtsystem muss die schnelle Übertragung von Prozessparametern und Eingriffsbefehlen bei gleichzeitiger Entkopplung von Echtzeit- und Nicht-Echtzeitsystem sein.

Zwischen Robotersteuerung und Echtzeitsystem soll daher ein Bussystem verwendet werden, über das Positionsdaten effektiv und schnell übertragen werden können. Zwischen Echtzeitsystem und Server des Benutzerunterstützungssystems muss ein Protokoll implementiert werden, das zu existierenden Netzwerkstrukturen kompatibel ist. Es soll hier ein einfaches, hochperformantes Protokoll entwickelt werden, das auf dem TCP/IP-Standard basiert.

Zwar bietet UDP (User Datagram Protocol) als alternatives Protokoll der IP-Familie prinzipbedingt eine höhere Datendurchsatzrate, es verzichtet jedoch auf eine zuverlässige Verbindung, wie TCP sie bietet. UDP als verbindungslose, unidirektionale Kopplung stellt nicht den Empfang eines Datenpaketes sicher. Im Gegensatz dazu arbeitet TCP bidirektional und garantiert die korrekte Reihenfolge der Datenpakete.

Die Echtzeitplattform soll damit in der Lage sein, vom Anwender spezifizierte Roboterprogramme ablaufen zu lassen und dabei gleichzeitig aktuelle Informationen aus dem Prozess, z.B. als Grundlage für eine Echtzeitsimulation, zu liefern.

Diese Schnittstelle soll auch für die echtzeitnahe Integration von Sensorinformationen in den laufenden Prozess dienen. Dieser Ansatz ermöglicht z.B. die Online-Bahnkorrektur auf Basis von Sensorinformationen, Interaktionen des Anwenders oder eine Prozesskorrektur auf Initiative der mitlaufenden Online-Simulation.

5.2.2.4 Prozessanalyse und Evaluierung

Die Analyse des Fertigungsprozesses soll anhand zweier Ansätze realisiert werden. Einerseits sollen von Maschinen aufgezeichnete Produktionsdaten ausgelesen und integriert werden. Anschließend kann eine Bewertung der ausgeführten Arbeiten im

Vergleich mit der Planung ausgeführt werden. Die Anbindung der von den Maschinen generierten Daten wie z.B. Schweißleistung/Stunde erfolgt ähnlich wie die bereits beschriebene Integration der Prozessdaten durch die Echtzeitplattform.

Andererseits soll die Prozessevaluierung auf Basis von geplanten und realen Prozesszeiten durch ein innerhalb des Clients ablaufendes Modul vorgenommen werden. Dieses Modul soll die in den Serverdatenbanken abgelegten Zeitpläne auslesen, diese mit den tatsächlich erreichten Zeiten vergleichen und hieraus eine grafische Darstellung generieren. Die grafische Auswertung kann vom Anwender im Client konfiguriert werden.

5.2.2.5 Prozessüberwachung und -regelung

Das System soll durch eine Prozessüberwachung und -regelung ergänzt werden. Diese Funktionen dienen der Integration von realen Prozesszuständen mit dem Ziel, Interaktionsmöglichkeiten zu schaffen. Gleichzeitig ist eine teilautomatische Regelung des Prozessablaufs zu entwickeln.

Das Modul koppelt sich mithilfe der Echtzeitplattform in den Fertigungsprozess ein. Es verfügt somit über alle relevanten Prozessdaten und kann diese im Client visualisieren.

Die in den Client ausgelagerten Module basieren auf der in Abbildung 5-2 gezeigten Architektur. Die Module sollen vollständig innerhalb des Clients ablaufen und eine beliebige Anzahl von Parametern einkoppeln können. Gleichzeitig sollen beliebig viele Clients gleichzeitig Zugriff auf die Daten haben. Ein entsprechend auszuwählender Middlewarestandard soll diese Anforderungen erfüllen.

5.2.2.6 Kommunikation

Das in das Unterstützungssystem zu integrierende Kommunikationsmodul soll den Informationsaustausch der Werker untereinander, aber auch der Abteilungen des Unternehmens fördern.

Um externen Kommunikationspartnern den Zugriff auf die Informationen zu ermöglichen, sollten die Kommunikationsmechanismen soweit wie möglich konform mit anerkannten Industriestandards sein. Für persönliche Informationen bietet sich die Nutzung von in die Benutzeroberfläche integrierten E-Mail-Diensten an.

Zusätzlich soll ein Modul implementiert werden, dass interaktive Diskussionen zu vom Anwender zu bestimmenden Themen ermöglicht. Das Konzept sieht eine vollständige Integration dieses Moduls in die Client-Server-Architektur vor.

Der Zugriff auf die Diskussionsforen ist damit unabhängig von Ort und technischer Plattform des Kommunikationsteilnehmers. Beispielsweise können Notizen eines

Werkers über Probleme bei der Fertigung eines bestimmten Werkstücks direkt von der Konstruktionsabteilung gelesen und auch kommentiert werden. Auch andere Worker können zur Lösung des Problems beitragen oder auch die verfügbaren Informationen bei ähnlichen Problemen nutzen.

Eine zusätzlich implementierte Volltext-Suchfunktion ermöglicht das schnelle und einfache auffinden der gewünschten Informationen.

6 Applikation und Integration

Die in den vorhergehenden Kapiteln erarbeitete Architektur für ein verteiltes Benutzerunterstützungssystem wurde in der exemplarischen Implementierung und Integration in zwei Werkstätten einer Schiffswerft verifiziert. Basierend auf der Analyse der allgemeinen Anforderungen der Werker an ein solches System wurde die generische Architektur spezifiziert, die als Ausgangspunkt für die unterschiedlichen Implementierungen galt. Beide Produktionsstätten verwenden grundsätzlich dasselbe Basissystem, jedoch unterscheiden sich die Systemderivate im Bereich der vermittelten Inhalte und Funktionalitäten. Somit können anhand dieser Beispiele die hohe Skalierbarkeit und die Adaptivität des Ansatzes demonstriert werden.

Dieses Kapitel beschreibt die Konzipierung und Umsetzung der erarbeiteten Systemspezifikationen aus technischer Sicht.

6.1 Modulares Gesamtsystem

Die folgenden Abschnitte beschreiben die Teilmodule des im Zuge dieser Arbeit entwickelten verteilten Produktionsunterstützungssystems DIS.MODUS (**Distributed Modular User Support System**). Das System bietet folgende Kernfunktionalitäten, die, wie in Abbildung 6-1 gezeigt wird, integriert wurden:

- ➔ Benutzerorientierte Informationssynthese
- ➔ Echtzeitfähige Prozessintegration
- ➔ Programmierung und Simulation
- ➔ Informationssystem
- ➔ Integrierte Planung
- ➔ Prozessanalyse und Evaluierung
- ➔ Prozessüberwachung und -regelung
- ➔ Kommunikation

Auf die verschiedenen Teilmodule wird in den folgenden Kapiteln ausführlich eingegangen.

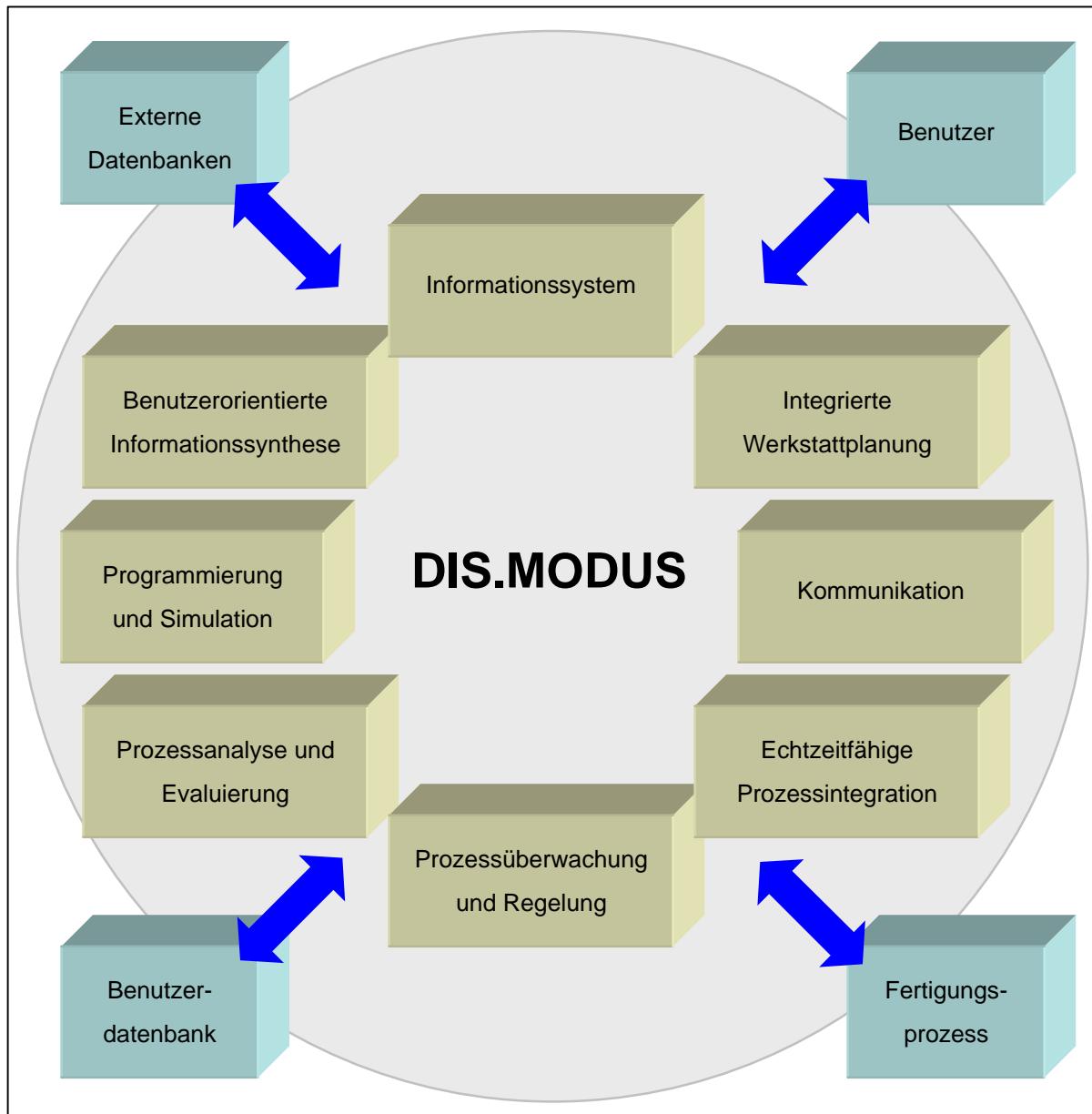


Abbildung 6-1: Kernfunktionalitäten des Fertigungsunterstützungssystems
DIS.MODUS

6.2 Datentechnische Infrastruktur

6.2.1 Serversystem

Das Produktionsunterstützungssystem basiert auf einer Client-Server-Architektur. Der Server dient – im Gegensatz zu herkömmlichen Client-Server-Systemen – hier nicht der reinen Speicherung von Daten und deren Organisation, sondern übernimmt gleichzeitig die Rolle einer Integrationsplattform für die verschiedenen Module.

Als Serversystem wurde aus den in Kapitel 4.5 beschriebenen Gründen Lotus Domino gewählt. Der Server übernimmt die folgenden Aufgaben:

- ➔ Statischer und dynamischer Webserver
- ➔ Server für eingebettete aktive Module
- ➔ Benutzeroauthentisierung
- ➔ Benutzerkontenverwaltung
- ➔ Informationssynthese in Abhängigkeit vom Benutzer
- ➔ Generische Kommunikationsdienste
- ➔ ODBC-Server
- ➔ Verwaltung von Referenzen

Zusätzlich zu den aufgeführten Funktionen wurde der Server um Dienste erweitert, von denen einige innerhalb der Domino-Umgebung ausgeführt werden und andere autonom ablaufen.

6.2.2 Client

Die Architektur sieht auf der Seite des Clients einen Web-Browser vor. Diese Lösung ermöglicht den Einsatz verschiedener Technologien für die Integration von aktiven Modulen.

Da der Microsoft Internet Explorer 5 sehr detaillierte Konfigurationsoptionen für die Zugriffsrechte der auf den Client übertragenen Programme besitzt, sollte dieser für den vollen Funktionsumfang des Unterstützungssystems verwendet werden. Somit ist es zum Beispiel möglich, dass authentisierte Java-Applets auf das lokale Dateisystem des Clients zugreifen und so die Systemkommunikation stark vereinfachen.

Die verschiedenen Ansätze für die Integration von Applikationen in den Client, die in diesem System realisiert wurden, werden nun dargestellt.

6.2.2.1 Integration durch Zentralisierung

Unter Zentralisierung wird die Verlagerung einer Problemlösung vom Client auf den Server verstanden. Eine Applikation wird dann nicht mehr auf dem Client ausgeführt, sondern befindet sich zentral auf dem Server und erstellt auf Anforderung oder in bestimmten Zeitintervallen HTML-Dokumente, die vom Anwender abgerufen werden können. Im wesentlichen sind folgende Vorteile mit der Zentralisation verbunden:

- Zentrale Installation, Administration und Konfiguration: Eine Applikation muss nur einmal auf dem Server installiert und konfiguriert werden. Die Administration kann zentral erfolgen.
- Einsparung von Systemressourcen der Clients: Die Applikation muss nicht auf jedem Client gespeichert werden und spart somit Speicher. Die Hardware der Clients kann kostengünstiger ausgelegt werden.
- Höherer visueller Integrationsgrad: Für den Anwender erscheint die serverseitig integrierte Applikation direkt im Web-Browser und nicht in einem separaten Fenster. Für ungeübte Anwender ist die Bedienung so einfacher.
- Höhere Sicherheit: Der rein weborientierte Zugriff auf eine serverseitige Applikation schirmt den Anwender vom System ab. Diese Verbindung erlaubt nur authentifizierten Nutzern den Zugriff auf diese Applikation.
- Höhere Stabilität: Der weborientierte Zugriff auf die Applikation erlaubt nur klar definierte Interaktionen. Das System ist so vor Fehlbedienungen, z.B. dem Herunterfahren des Rechners ohne ein vorheriges Ausloggen aus einer Datenbank, geschützt.

Die Zentralisation bringt jedoch auch Nachteile mit sich, die alternative Integrationsmethoden erfordern:

- Stärkere Belastung des Servers: Die Ressourcenauslastung des Servers wird z.B. bei gleichzeitigem Zugriff mehrerer Clients auf eine Applikation stark erhöht und darf unter keinen Umständen ein kritisches Maß erreichen, da dann die rechtzeitige Ausführung von Systemfunktionen nicht mehr gewährleistet werden kann.
- Performanceintensive Applikationen nicht möglich: Bestimmte Applikationen erfordern ein hohes Maß an Performance. Beispielsweise muss die 3D-Visualisierung von Werkstücken lokal auf einem Client ausgeführt werden, da für eine serverseitige Ausführung weder Netzwerkdurchsatz noch Rechenleistung des Servers ausreichen würden. Der Server verwaltet lediglich die 3D-Modelle, die zum Client übertragen und in einem VRML-Client oder in einer OpenGL-Applikation dargestellt werden.

- ➔ Eingeschränkte Nutzerinteraktion: Durch die Abschirmung des Nutzers wird auch seine Interaktionsfähigkeit eingeschränkt. Mögliche Bedienelemente werden auf das innerhalb des HTML-Standard Mögliche begrenzt [W3H00].

Ein Beispiel für die Integration durch Zentralisierung ist das Kommunikationsmodul, auf das in Kapitel 6.3.5 näher eingegangen wird.

6.2.2.2 Integration durch Java

Eine weitere Möglichkeit der Integration von Applikationen besteht in der Anwendung von Java. Durch den Einsatz dieser Sprache wird die Ausführung des Programms innerhalb des Web-Browsers möglich. Das Java-Applet wird dabei vom Server in HTML-Dokumente eingebettet und zum Client heruntergeladen.

Für viele Applikationen stellt diese Technologie eine sehr interessante Lösung dar. Nachteile dieses Ansatzes sind unter anderem die relativ geringe Ausführungsgeschwindigkeit von Java-Applikationen und der aufgrund der systemimmanenten Sicherheitsfunktionen von Java eingeschränkte Zugriff auf beliebige Netzwerkadressen bzw. Datenbanken. Java erlaubt prinzipiell lediglich den Zugriff auf den Server, von dem das Applet heruntergeladen wurde.

Diese Integrationsform wurde für die Visualisierung und Verifikation von Prozessparametern gewählt.

6.2.2.3 Integration lokaler und nicht-lokaler Anwendungen

Die Integration von bereits vorliegenden Anwendungen bzw. von Anwendungen, die nicht nach den vorher beschriebenen Methoden integriert werden können, erfordert die Entwicklung von neuen Verfahren. Es wird zwischen Applikationen, die auf dem Client-Betriebssystem lauffähig sind und Applikationen, die unter UNIX-Systemen ausgeführt werden müssen, unterschieden.

Integration von Windows-Applikationen durch WEB.LAP

Ausgangspunkt ist hierbei die Annahme, dass die Anwendung als ausführbare Datei vorliegt und unter dem Client-Betriebssystem ausführbar ist. Da davon ausgegangen werden muss, dass der Anwender wenig mit dem Betriebssystem vertraut ist und alle Funktionalitäten von DIS.MODUS innerhalb der systemeigenen Benutzeroberfläche aufrufbar sein sollen, musste eine Technik konzipiert und entwickelt werden, die den Programmstart direkt aus dem Web-Browser heraus ermöglicht.

Das Konzept sieht eine Java-Hilfsapplikation (WEB.LAP, **Web-based launch of Applications**) vor, die vom Serversystem zum Client übertragen wird und dort das gewünschte Programm startet. Voraussetzung ist hier die entsprechende

Konfiguration des Web-Browsers, um dem Java-Applet das Starten eines Programms zu ermöglichen. Das Prinzip ist in Abbildung 6-2 dargestellt.

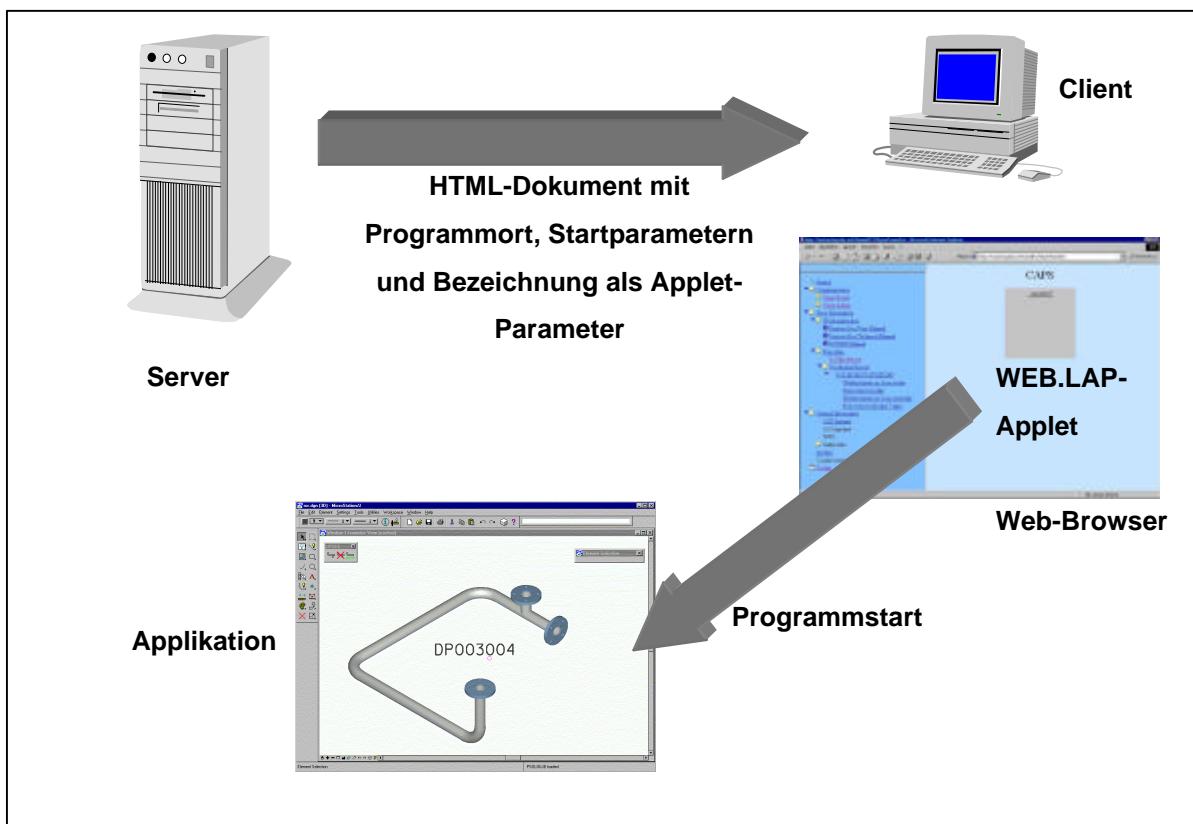


Abbildung 6-2: Integration durch WEB.LAP

Durch diese Methode wird die Wartbarkeit des Gesamtsystems erhöht. Sämtliche Konfigurationsparameter für den Programmstart sind in der HTML-Seite enthalten, die das Java-Applet aufruft. Diese Seite ist auf dem Server abgelegt und kann dort zentral administriert werden.

Ein weiterer Vorteil ist die generische Implementierung von WEB.LAP, dessen Kern, das Java-Applet, völlig unabhängig vom jeweiligen Einsatz ist und für jede Anwendung lediglich umkonfiguriert werden muss. Prinzipiell kann WEB.LAP somit für den Aufruf aller nicht direkt in das System integrierbaren Funktionen dienen.

Mit dieser Technik kann über das Netzwerk auch ein nicht auf dem Client gespeichertes Programm gestartet werden.

6.3 Systemkomponenten

6.3.1 Echtzeitfähige Integration des Fertigungsprozesses

Die enge Verknüpfung des Unterstützungssystems mit dem Fertigungsprozess erfordert eine echtzeitfähige Kopplung an die Prozessumgebung. Da das Unterstützungssystem aufgrund der erforderlichen Benutzerinteraktivität selbst nicht echtzeitfähig sein kann, muss eine Echtzeitplattform in das Systemkonzept integriert werden.

Das hier ausgewählte Echtzeitbetriebssystem WindRiver VxWorks 5.2 entspricht dem POSIX-Standard und wird somit den Forderungen nach Standardisierung und Erweiterbarkeit gerecht. Die Hardware-Plattform, ein PowerPC-System mit drei Prozessoren und VME-Bus-Architektur ermöglicht die enge Prozesskopplung durch die direkte Anbindung des Prozessrechners an die Steuerungskomponenten der Fertigungszelle.

Die zentrale Komponente der Echtzeitplattform, das Motorola MVME 1604-002 PowerPC-Board ist wie folgt spezifiziert:

- ➔ 100 MHz MPC604 PowerPC-Prozessor
- ➔ 16 MB DRAM
- ➔ 32-Bit VME-Bus-Interface
- ➔ 32-Bit PCI-Bus-Interface
- ➔ Ethernet-Interface
- ➔ SCSI-2-Interface
- ➔ SVGA-Grafik-Interface
- ➔ Eine parallele und vier serielle Schnittstellen

Die hohen Anforderungen an die Reaktionszeiten sollen durch die Kopplung dreier dieser Boards erfüllt werden. Die drei Prozessorboards sind über den VME-Bus an die folgenden Schnittstellen angebunden:

- ➔ Drei VIPC610-Industry-Pack Trägerplatten für 6 HE VME-Bus-System
- ➔ Ein IP-Precision ADC – 20-Kanal 12-Bit-A/D-Wandler
- ➔ Drei IP-DAC – 6-Kanal-D/A-Wandler
- ➔ Zwei IP-Digital48 – 48 digitale Ein-/Ausgänge
- ➔ Drei IP-quadrature – 4-Kanal-Quadratur-Decoder

Als Entwicklungsplattform können sowohl Windows- als auch UNIX-Rechnersysteme genutzt werden. Für beide Plattformen steht die Entwicklungsumgebung Tornado zur Verfügung.

Die Gesamtstruktur der Echtzeithardware ist in Abbildung 6-3 dargestellt.

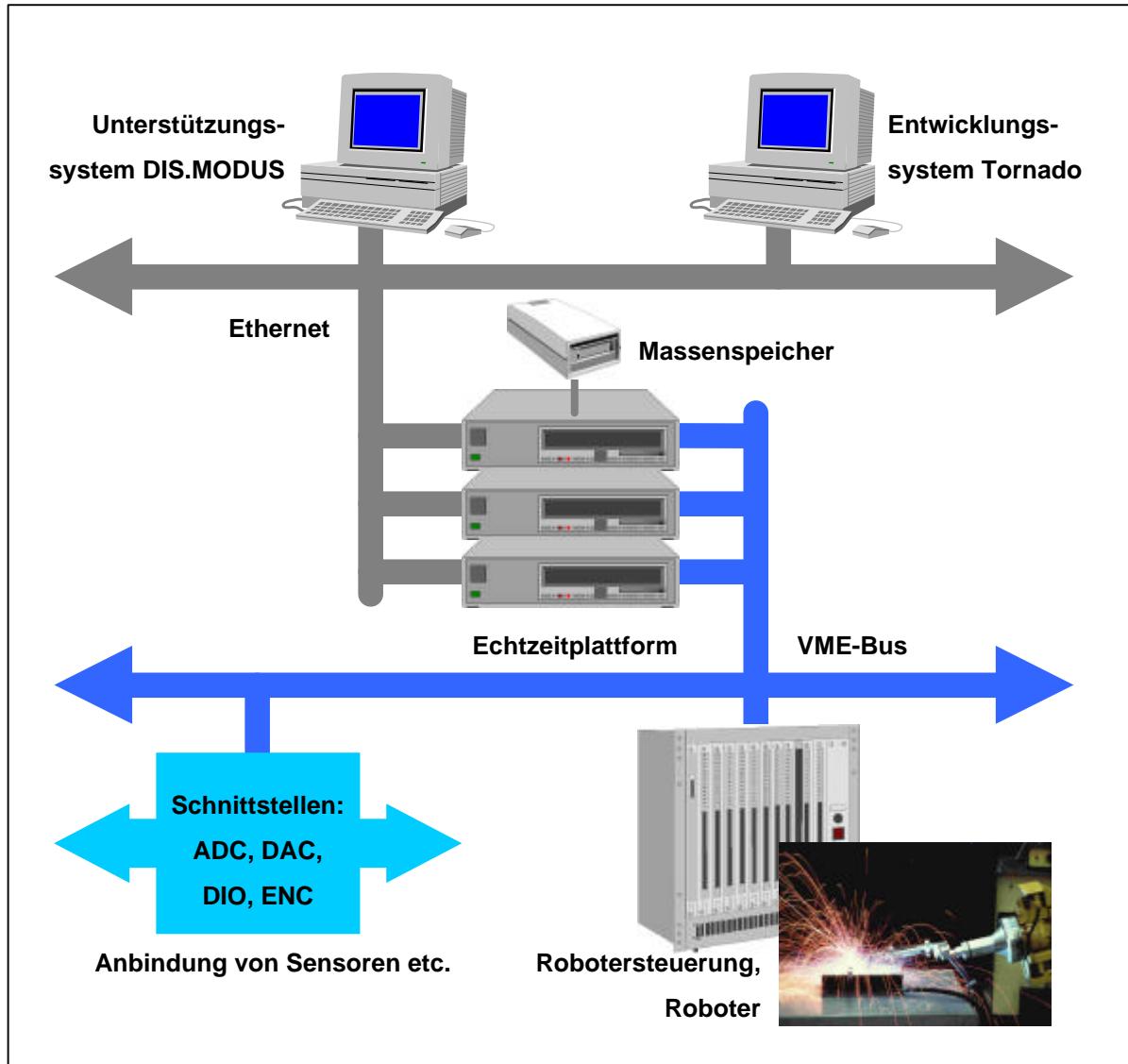


Abbildung 6-3: Struktur des Echtzeit-Teilsystems

Das zum Einsatz kommende Betriebssystem VxWorks unterstützt prioritätengesteuerte Multitasking- und Multiprozessor-Anwendungen. Zwischen parallel ablaufenden Tasks kann mit Hilfe von Message Passing über Message Queues kommuniziert werden. Zwischen den einzelnen Prozessoren der Plattform können zusätzlich mithilfe von Shared-Memory- und Semaphoren-Technik Daten ausgetauscht werden.

Im Rahmen dieser Arbeit wurde ein Softwarekonzept zur netzbasierten Ansteuerung und Echtzeitmanipulation von Industrierobotern entwickelt und an den hier zum

Einsatz kommenden Reis-Roboter adaptiert. Der Roboter wird hier über den sogenannten Dual-Ported RAM angekoppelt. Im Dual-Ported RAM legt die Robotersteuerung alle Positionsregister, also aktuelle Ist- und Soll-Achsstellungen sowie Ist- und Sollwerte der kartesischen Koordinaten ab [REI95]. Diese Positionsregister können durch die Kopplung mit der Echtzeitplattform ausgelesen und verändert werden (Abbildung 6-4). Die Querdurchsatzgeschwindigkeit des Dual-Ported RAM beträgt dabei nominell zwei Millisekunden und ist damit groß gegen den roboterspezifischen System- und Interpolationstakt von gemittelt 30 ms. In einem ersten Ansatz wurde davon ausgegangen, dass der Prozessrechner nach einer einmaligen Synchronisation mit einer Taktrate T_e von exakt 30 ms arbeiten soll und somit sichergestellt werden kann, dass jedem Robotertakt ein Takt des Interpolators zugeordnet werden kann.

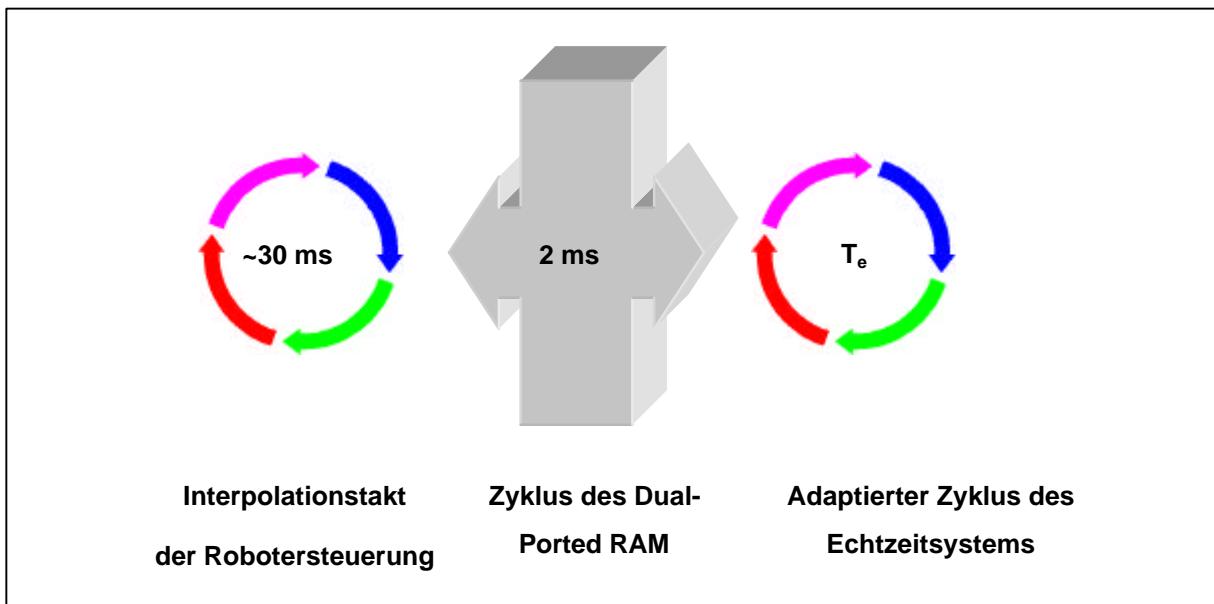


Abbildung 6-4: Zeitliche Kopplung des Echtzeitystems an den Interpolationstakt der Robotersteuerung

Untersuchungen des Zeitverhaltens der Kopplung des Prozessrechners ergaben jedoch, dass unter Umständen ein zeitlich divergierendes Verhalten der beiden Systeme zu beobachten ist, da die vom Prozessrechner aus unter realen Bedingungen gemessene Taktzeit nicht immer exakt 30 ms beträgt, sondern vielmehr eine Verteilung wie in Abbildung 6-5 dargestellt, gemessen werden konnte. Besonders problematisch ist die asymmetrische Streuung zu kürzeren Zeiten hin, die tendenziell die Synchronität stört.

Durch die Implementierung eines Synchronisationsalgorithmus konnte die Stabilität des Gesamtsystems stark verbessert werden. Realisiert wurde dieser Ansatz mit Hilfe eines jedem Takt vorgesetzten Wartezustands der Interpolation. Die Kommunikationsroutine wartet somit bei jedem Takt auf das Ready-Signal der

Robotersteuerung und schreibt erst dann einen neuen Positionswert in den Koppelspeicher.

Die Abweichungen der Robotergeschwindigkeit, die sich wegen der variablen Update-Routine in Kombination mit der Interpolation auf Basis des konstanten Zeittaktes von 30 ms ergeben, bewegen sich im Bereich von weniger als 2% und sind damit vernachlässigbar gering.

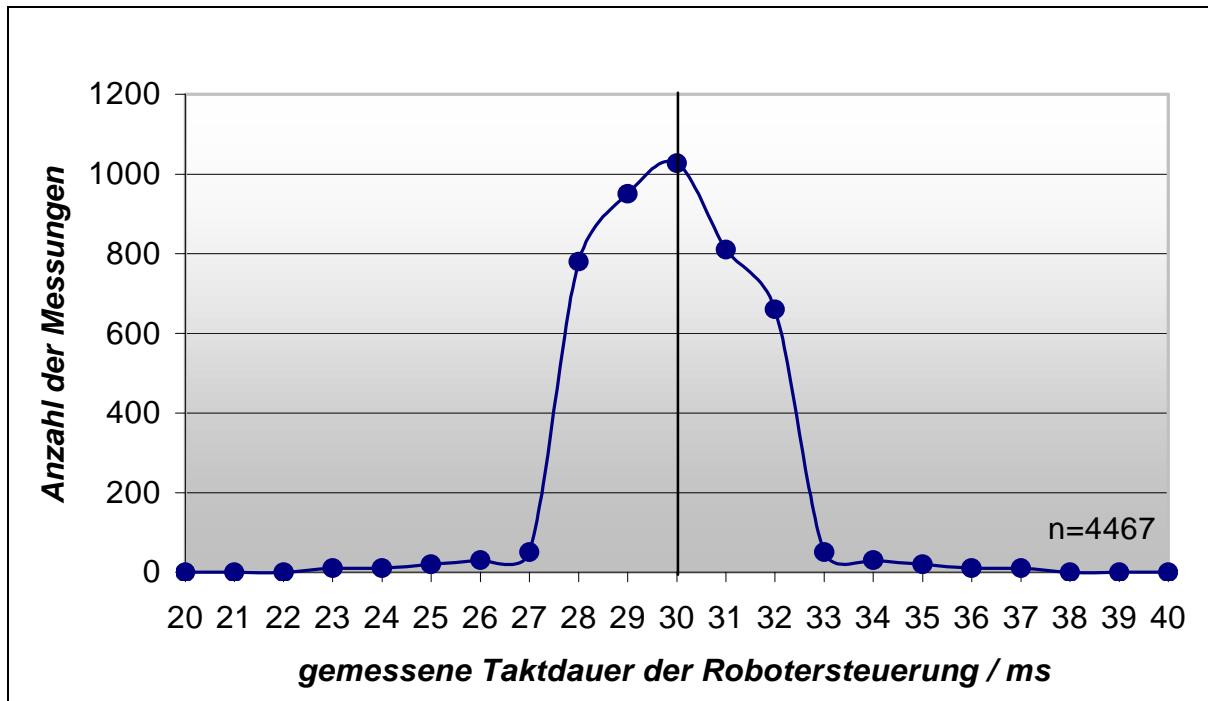


Abbildung 6-5: Verteilung des Interpolationstaktes der Robotersteuerung

Die für das Echtzeitsystem entwickelte Software verfügt über Algorithmen, die eine umfassende Steuerung des Roboters ermöglichen. Sie basiert auf folgenden in [MAT97] für eine modulare Steuerungsarchitektur konzipierten Modulen:

- ➔ Kommunikationsmodul für die Koppel-RAM-Anbindung der Robotersteuerung
- ➔ Hochgeschwindigkeits-Interpolationsalgorithmus für CP (Continuous Path) und PTP (Point-to-Point) Bewegungen
- ➔ Sensorintegrationsmodul

Diese Module wurden an die hier verwendete Robotersteuerung angepasst und durch folgende neue Komponenten ergänzt:

- ➔ Kommunikationsmodul für die Netzwerkanbindung des Echtzeitsystems
- ➔ Interpreter für lokal gespeicherte Roboterprogramme im Pseudocode

Diese neuen Komponenten erlauben die Anbindung des Unterstützungssystems zur Online-Simulation und Prozessinteraktion.

Im Folgenden wird die Arbeitsweise des Echtzeitmoduls beschrieben. Sie stellt die Grundlage für weitere Teilmodule dar.

Das Softwarekonzept des Echtzeitsystems basiert auf einer Gliederung der Teilmodule in drei parallel agierende Algorithmen. Die drei Paralleltasks erfüllen dabei die folgenden Aufgaben:

- ➔ Generierung der Bahnstützpunkte
- ➔ Transfer der Bahnstützpunkte zum Roboter
- ➔ Datenaustausch mit Hilfe der Netzwerkschnittstelle

Die drei Paralleltasks werden bei der Initialisierung des Systems gestartet und kommunizieren anschließend über Message-Queues unter Einsatz von Semaphoren, die zur Sicherstellung der Datenintegrität dienen.

Im ersten Schritt werden die Positionsdaten eines Roboterprogramms, das in einem Pseudo-Code auf der lokalen Festplatte des Prozessrechners vorliegt, eingelesen. Der generische Pseudo-Code beinhaltet alle für die Ausführung eines Roboterprogramms notwendigen Daten und ist so konzipiert, dass er unabhängig vom verwendeten Robotertyp verwendet werden kann. Die eingelesenen Positionsdaten werden anschließend vom Interpolationsmodul analysiert und auf der Zeitbasis der Robotersteuerung von 30 ms in entsprechende Bahnstützpunkte umgerechnet. Der Abstand der Stützpunkte variiert demzufolge mit der jeweiligen Geschwindigkeit des Roboterwerkzeugs auf diesem Teilstück der Bahn. Die Interpolation berücksichtigt dabei sowohl die für die Bahn definierte Geschwindigkeit, als auch die in der Nähe des Anfangs- bzw. Endpunktes der Bahn zu berücksichtigenden Beschleunigungs- und Abbremsphasen unter Berücksichtigung der jeweils gültigen Überschleifparameter [MAT97].

Die so generierten Stützpunkte werden anschließend mit Hilfe einer Message-Queue an den Prozess zur Kommunikation mit der Robotersteuerung übergeben. Dieser hoch priorisierte Prozess überträgt die Stützpunkte dann sequenziell in den Koppel-RAM, der von der Robotersteuerung als Positionsregister genutzt wird. Die Kommunikationsmechanismen zwischen den separaten Tasks müssen dabei so realisiert sein, dass die korrekte Reihenfolge und Vollständigkeit der Stützpunktsequenzen zu jedem Zeitpunkt garantiert wird. Zu diesem Zweck wurden Semaphoren eingeführt, die das Beschreiben einer Message-Queue erst dann ermöglichen, wenn der auslesende Prozess einen oder mehrere Werte abgeholt hat und somit Speicherplatz in der Queue frei geworden ist.

Die auf dem TCP/IP-Protokoll basierende Netzwerkschnittstelle öffnet das Echtzeitsystem einer Vielzahl von Applikationen, die über diese Schnittstelle echtzeitnahe Positionsdaten des Roboters abrufen oder seinen Bewegungsablauf manipulieren können. In dieser Arbeit wurden folgende exemplarische Anwendungen dieser Schnittstelle realisiert:

- Anbindung eines Online-Simulationssystems
- Simulationsbasierte Benutzerinteraktion
- Sensorintegration zur Online-Prozessoptimierung

Die Netzwerkanbindung der Roboteransteuerung muss zu vielen Systemen kompatibel und innerhalb unterschiedlicher Infrastrukturen einsetzbar sein. Gleichzeitig sind die Anforderungen an Reaktionsgeschwindigkeit und Übertragungsleistung des Protokolls hoch. Das entwickelte Protokoll basiert auf dem TCP/IP-Standard und ermöglicht den Austausch der in folgender Tabelle dargestellten Parameter:

Datensatz	Parameter
Übertragung von Daten vom Echtzeitsystem zu externer Applikation:	
Kartesische Roboterposition, Orientierung	$\vec{X} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}, \quad \vec{O} = \begin{pmatrix} A \\ B \\ C \end{pmatrix}$
Position in Roboterkoordinaten (Achswinkel)	$\vec{R} = \begin{pmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \\ \mathbf{q}_4 \\ \mathbf{q}_5 \\ \mathbf{q}_6 \end{pmatrix}$
Übertragung von Daten von externer Applikation zum Echtzeitsystem:	
Bahnkorrekturvektor	$\Delta \vec{X} = \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}$
Inkrementelle Bewegungsanforderung	$\vec{X}_x = \begin{pmatrix} i_x \\ 0 \\ 0 \end{pmatrix}, \quad \vec{X}_y = \begin{pmatrix} 0 \\ i_y \\ 0 \end{pmatrix}, \quad \vec{X}_z = \begin{pmatrix} 0 \\ 0 \\ i_z \end{pmatrix}$

Anpassung des Inkrement	$\Delta \vec{I} = \begin{pmatrix} \Delta i_x \\ \Delta i_y \\ \Delta i_z \end{pmatrix}$
Nummer des zu ladenden Programms	P

Tabelle 6-1: Mögliche Kommunikationsvarianten des Netzwerkinterfaces

Das Echtzeitsystem stellt externen Applikationen die aktuelle Roboterposition zur Verfügung. Die Übertragung erfolgt auf Anfrage der Applikation. Übertragen werden dabei jeweils die kartesischen Koordinaten als auch die Position in Roboterkoordinaten. Tests im typischen Werkstattumfeld mit einer 10 MBit-Netzwerkanbindung und einem zwischengeschalteten Hub (Verteiler) ergaben, dass die Reaktionszeit der Schnittstelle so gering ist (Tabelle 6-2), dass die echtzeitnahe Verfolgung z.B. eines Programmablaufs ermöglicht wird. Die Messungen wurden jeweils unter normaler Netzlast, d.h. während des laufenden Betriebes in der Werkstatt, sowie unter sehr starker Netzlast durchgeführt. Die starke Netzlast wurde durch die parallel laufende Übertragung einer großen Datenmenge auf denselben Netzwerkabschnitt simuliert.

Paketgröße	Minimum	Durchschnitt	Maximum
Normale Netzlast			
24 Byte	1,473 ms	1,603 ms	1,876 ms
72 Byte	1,113 ms	1,587 ms	1,789 ms
136 Byte	1,512 ms	1,610 ms	1,989 ms
Starke Netzlast			
24 Byte	2,732 ms	7,744 ms	14,441 ms
72 Byte	3,580 ms	9,010 ms	16,451 ms
136 Byte	3,287 ms	8,337 ms	13,592 ms

Tabelle 6-2: Übertragungszeiten von Positionsdaten. Die Paketgröße eines Positionsdatensatzes liegt zwischen 24 und 136 Bytes

Die Orientierungsparameter werden im Euler-Format übertragen. Zusätzlich kann der aktuelle Zustand des Roboters (Bewegung aktiv / nicht aktiv) abgefragt, und Programme können gestartet bzw. beendet werden.

In der anderen Richtung ermöglicht die Netzwerkschnittstelle die direkte, echtzeitnahe Interaktion von externen Modulen, die autonom agieren bzw. vom Anwender initiiert Aktionen auslösen.

6.3.1.1 Autonome Prozessregelung durch integrierte Sensorik

Der Roboter kann mithilfe eines frei zu definierenden Inkrementvektors in kartesischen Koordinaten verfahren werden. Unter Verwendung dieser Funktionalität wurde ein Bahnkorrekturverfahren eingeführt, das während eines Programmablaufes die Positionsdaten der Roboterkinematik manipuliert. Diese Schnittstelle wird zur vollautomatischen Verfolgung einer Schweißnaht mit Hilfe eines optischen Sensors verwendet. Die Schnittstelle erlaubt dabei eine Bahnkorrektur in X-, Y- und Z-Richtung.

Die Verknüpfung des Korrekturvektors mit der geplanten Bahn erfolgt nach dem in Abbildung 6-6 gezeigten Prinzip. Die schematische Darstellung zeigt die interpolierten Teilstücke stark vergrößert.

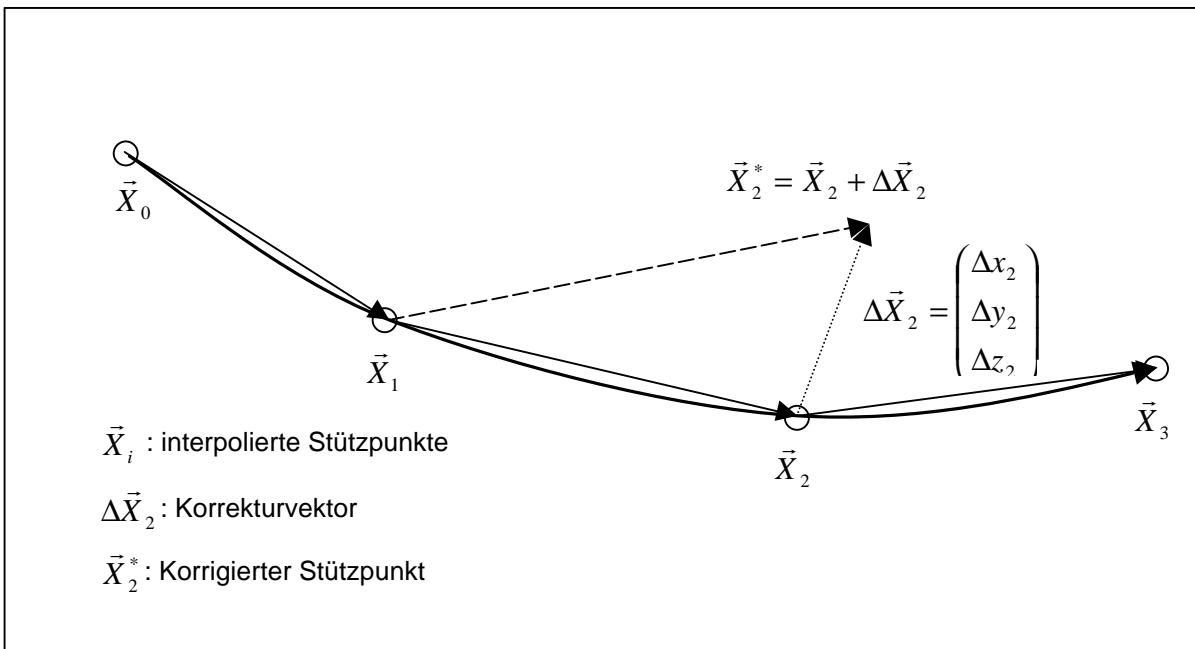


Abbildung 6-6: Korrektur einer interpolierten Bahn

Die Möglichkeit der Bahnkorrektur in X-, Y- und Z-Richtung erlaubt die präzise Bearbeitung von Freiformflächen und beliebig geformten Schweißnähten.

Im Folgenden wird eine Anwendung der Echtzeitplattform unter dem Aspekt der autonomen Prozessregelung aufgezeigt. Mit Hilfe der Integration eines 3D-Sensorsystems soll die Programmierung des Roboters vereinfacht und die Qualität der Schweißnaht verbessert werden.

Das hier verwendete Sensorsystem wurde speziell für die Anwendung in der Schweißtechnik entworfen. Orientiert an den verfahrensspezifischen Bedingungen des Schweißens stellt der Sensor Informationen für die Online-Korrektur der Werkzeugposition in Relation zum Werkstück zur Verfügung. Die

Schweißbearbeitung z.B. von frei geformten Blechen im Schiffbau ist nicht nur aufgrund einer möglicherweise komplexen Nahtgeometrie schwierig, sondern kann durch den Einfluss von Fertigungstoleranzen und Wärmeverzug zur Qualitätsminderung der Schweißnaht oder im Extremfall zum Auseinanderlaufen von Brennerbahn und Naht führen [SCW95].

Der Sensor verwendet ein Lichtschnittverfahren mit Steifenprojektion zur Bestimmung von Nahtgeometrie, der Lage der Naht und der relativen Brennerorientierung. Im Gegensatz zu den sonst üblichen Verfahren der Lichtstreifenprojektion, bei denen lediglich Abstand, Versatz und Neigung des Brenners bestimmt werden können, verwendet dieser Sensor drei Lichtstreifen, die eine Bestimmung aller fünf Freiheitsgrade ermöglichen. In Verbindung mit dem Handhabungssystem kann somit die Position des Brenners in allen drei Raumrichtungen korrigiert werden. Prinzipiell ist auch die Korrektur der Brennerdrehung und -neigung möglich, jedoch wurde in dieser exemplarischen Implementierung darauf verzichtet, da Korrekturen hauptsächlich in den drei Raumrichtungen notwendig sind und der Aufwand für eine winkelbezogene Korrektur aufgrund der schwierigen Rückwärtstransformation sehr groß wäre.

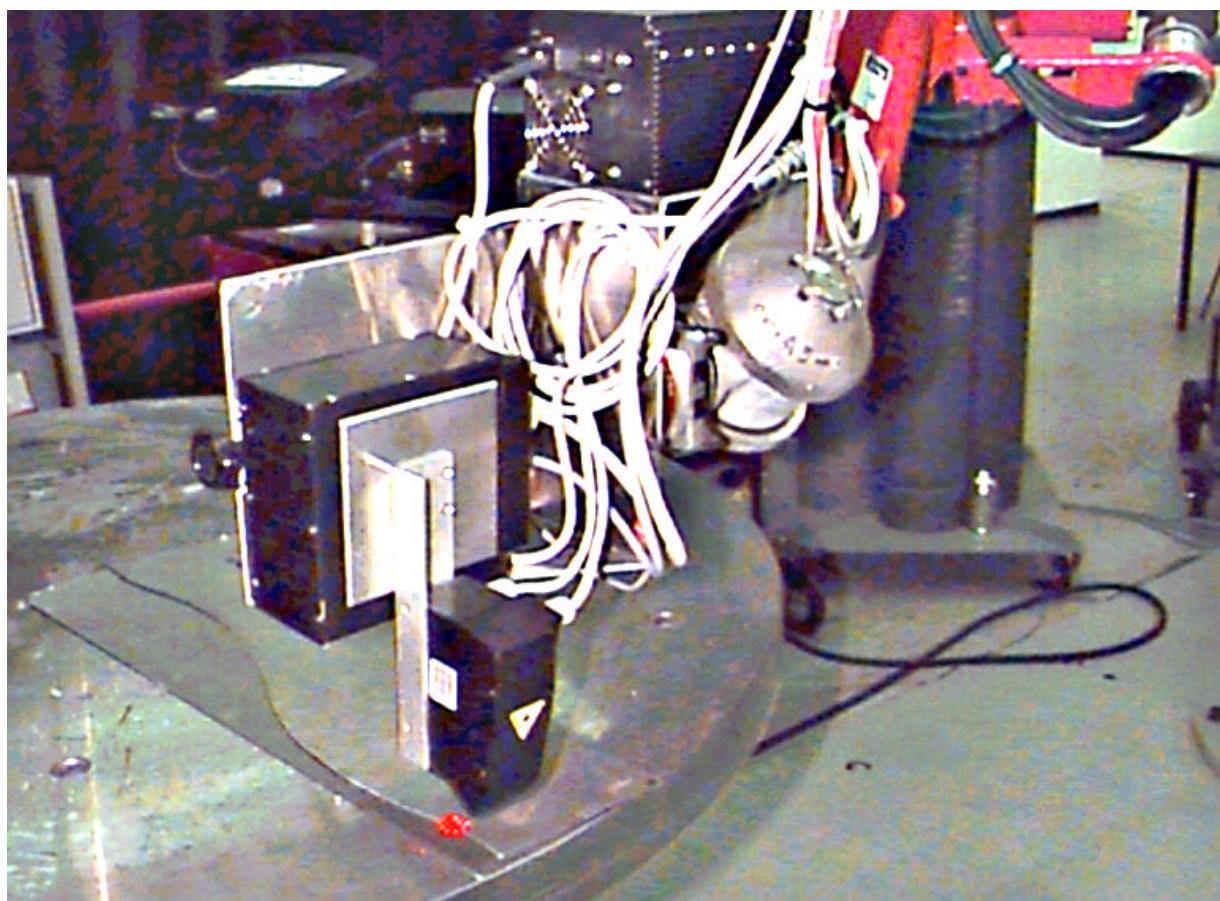


Abbildung 6-7: Integriertes Sensorsystem zur Nahtverfolgung

In Abbildung 6-7 ist der am Roboterhandgelenk montierte Sensor mit seiner charakteristischen Lichtstreifenprojektion zu erkennen. Bei dem Werkstück handelt es sich um zwei Bleche, die durch eine Überlappnaht zusammengeschweißt werden sollen. Die Programmierung des Roboters sieht eine geradlinige Bahn vor, jedoch folgt der Roboter aufgrund der integrierten Sensorik der stark deformierten Naht.

Bei der Integration des Sensors sorgt der in Abbildung 6-8 gezeigte Regelkreis für ein hinreichend stabiles Verhalten bei einem gleichzeitig großen Regelintervall. Die Verwendung eines einfachen P-Reglers hat sich in Testläufen mit verschiedenen Werkstücken als ausreichend herausgestellt. Durch die schnelle Nachführung des Sensors durch den Roboter wird erreicht, dass beim nächsten Stützpunkt idealerweise der Korrekturvektor gleich Null ist, sodass wieder mit dem gesamten vom Sensor erfassten Bereich auf neue Bahnabweichungen reagiert werden kann.

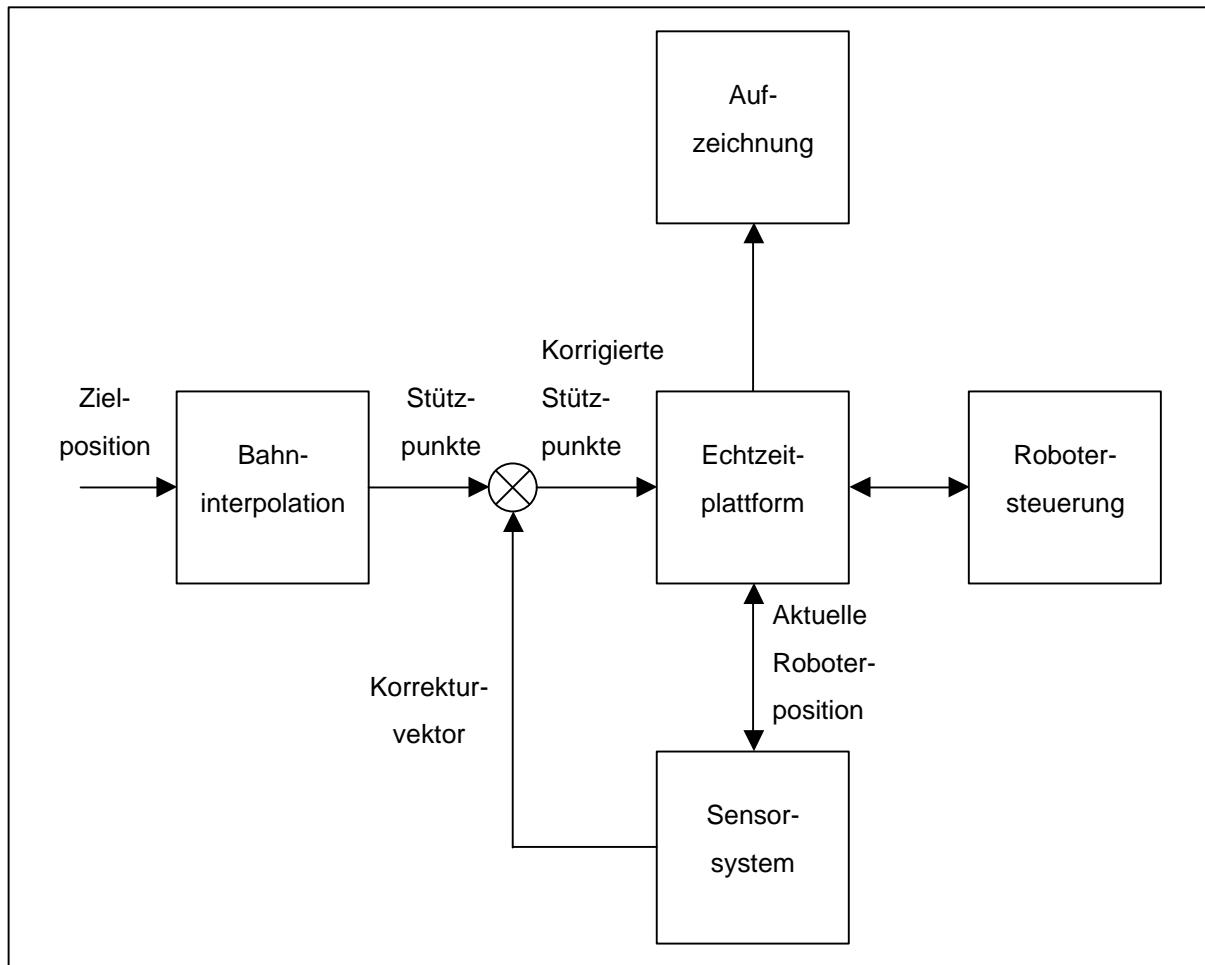


Abbildung 6-8: Regelkreis aus Echtzeitplattform und Sensor

Die maximale sensorbedingte Abweichung von einer programmierten Schweißbahn $\vec{\Delta X}$ berechnet sich aus der Summe der maximalen Korrekturvektoren $\vec{\Delta x}$ entlang der in n Teilstücke der Länge Δl unterteilten Bahn der Länge l :

$$\Delta \vec{X} = \sum_{i=0}^n \Delta \vec{x} \quad (6-1)$$

Da $\Delta \vec{x}$ über alle n konstant ist, ergibt sich

$$\Delta \vec{X} = n \cdot \Delta \vec{x} = \frac{l}{\Delta l} \cdot \Delta \vec{x} \quad (6-2).$$

Durch einsetzen von $\Delta l = v_{TCP} \cdot T_I$ erhält man

$$\Delta \vec{X} = \frac{l}{v_{TCP} \cdot T_I} \cdot \Delta \vec{x} \quad (6-3)$$

mit v_{TCP} als Geschwindigkeit des Roboter-TCP und T_I als Interpolationstakt der Robotersteuerung.

$\Delta \vec{x}$ entspricht der maximalen sensorbedingten Korrektur, also bei dem hier verwendeten Streifenprojektionssystem der halben Projektionsstreifenbreite \vec{s} :

$$\Delta \vec{X} = \frac{l}{2 \cdot v_{TCP} \cdot T_I} \cdot \vec{s} \quad (6-4)$$

Bei einem Messbereich des Sensors von 40 mm quer zur Schweißbahn, einer TCP-Geschwindigkeit von 100 mm/s und dem steuerungsbedingten Interpolationstakt von 30 ms ergibt sich so beispielsweise eine maximale sensorbedingte Korrektur der programmierten Bahn von 0,66 m quer zur Bewegungsrichtung bezogen auf einen Meter Bahn. Damit ist das Regelungssystem in der Lage, auch sehr starke Abweichungen einer Schweißbahn auszugleichen bzw. die Bearbeitung gekrümmter Bahnen wie z.B. Bögen oder Freiformflächen ohne explizite und aufwendige Programmierung zu ermöglichen. Zur autonomen Bearbeitung einer gekrümmten Schweißbahn reicht somit die Angabe des Startpunktes und des Endpunktes der Bahn aus, wenn gewährleistet werden kann, dass das Stabilitätskriterium des P-Reglers 6-4 eingehalten wird.

Das Sensorsystem wird von der Online-Schnittstelle bei jedem Systemtakt über die aktuelle Roboterposition informiert, sodass hier eine Verifikation der ermittelten Korrekturwerte erfolgen kann. Ist das Sensorsystem adaptiv ausgelegt, kann es das Korrekturverhalten des Roboters und der Robotersteuerung erkennen und seine Korrekturparameter dementsprechend anpassen. Im hier gewählten Beispiel war der Sensor zusätzlich mit einer dreiachsigen Verstelleinheit ausgerüstet, sodass dieser auf Korrekturen, die der Roboter aus Stetigkeitsgründen nicht schnell genug ausführen konnte, reagieren konnte und den Sensor entlang der Bahn geführt hat. Der so zustande gekommene Versatz des Sensors gegenüber dem Roboterwerkzeug wurde mit dem nächsten Korrekturvektor verrechnet, sodass sich insgesamt ein auch unter schwierigen Bedingungen und hohen Geschwindigkeiten sehr stabiles Gesamtsystem ergab.

Die Bahnverfolgung mithilfe des Sensors konnte erfolgreich um die Online-Kollisionskontrolle ergänzt werden. Durch diese Kombination konnte ausgeschlossen werden, dass der Roboter auf seiner nicht implizit vorherbestimmten Bahn mit Hindernissen wie z.B. Spannvorrichtungen kollidiert. Dabei wurden die aus der Bahnkorrektur berechneten Roboterkoordinaten in diesem experimentellen Aufbau über die Netzwerkschnittstelle an das Online-Simulationssystem WEB.MOVE gesendet, die beim Unterschreiten eines Mindestabstands zweier Objekte ein Stoppsignal generierte. Die in diesem Aufbau für kollisionsrelevante Daten verwendete Netzwerkschnittstelle kann nur zu Testzwecken dienen, da in der industriellen Anwendung nicht garantiert werden kann, dass zu jedem Zeitpunkt ausreichend Netzwerkkapazität zur Verfügung steht. Aufgrund der modularen Systemarchitektur kann prinzipiell hier auch ein echtzeitfähiges Bussystem eingesetzt werden.

6.3.1.2 Verteilte parameterbasierte Prozessvisualisierung und -interaktion

Im vorhergehenden Kapitel wurde ein Werkzeug für die Überwachung und Regelung der Kinematik einer automatisierten Fertigungseinheit vorgestellt. Die untersuchten schweißtechnischen Fertigungsprozesse sowohl in der Rohrwerkstatt als auch beim Zusammenbau der Panels in der Montagewerkstatt erfordern jedoch die Einbeziehung von weiteren relevanten Daten für eine umfassend prozessunterstützende Lösung. Auf Basis der in Kapitel 3 erarbeiteten Konzepte wurde daher eine in das Unterstützungssystem integrierbare Lösung angestrebt, die in der Lage ist, nicht orts- bzw. plattform- oder kontextgebunden Parameter eines Fertigungsprozesses für die verteilte Visualisierung und Regelung aufzubereiten und mit ihnen zu interagieren. Parameter sind hier beispielweise Schweißspannung, Schweißstrom oder auch der aktuelle Betrag des in Kapitel 6.3.1.1 beschriebenen sensorbasierten Korrekturvektors.

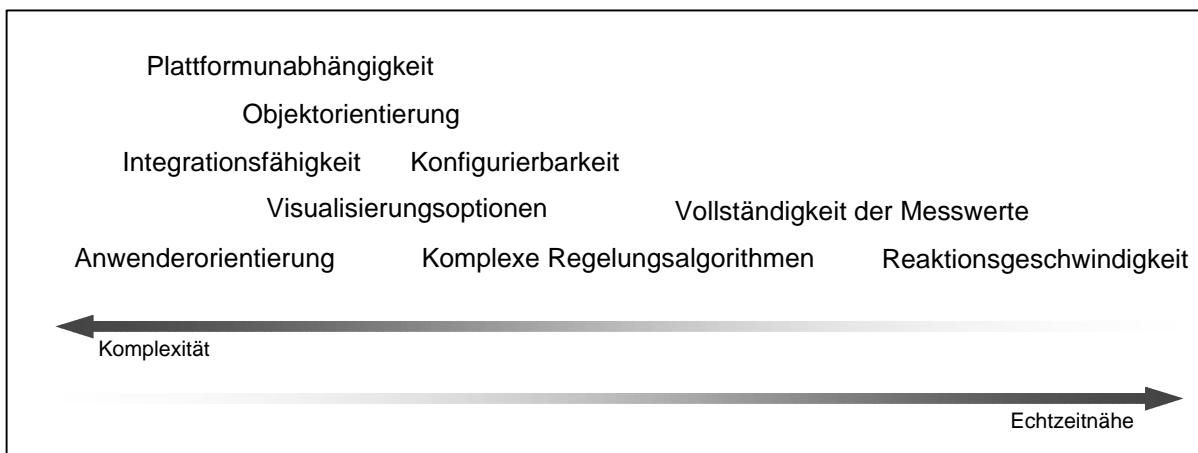


Abbildung 6-9: Spannungsfeld der Anforderungen an eine verteilte parameterbasierte Prozessregelung

Bei der Entwicklung des Systems wurde davon ausgegangen, dass die Anforderungen an eine solche Technologie stark mit dem jeweils betrachteten Parameter variieren. Die Schwerpunkte der Anforderungen sind im mittleren Bereich des in Abbildung 6-9 aufgezeigten Spannungsfeldes lokalisiert. Das Spannungsfeld resultiert aus den in [GAL95] aufgezeigten Kommunikationsanforderungen (Abbildung 5-4) für die automatisierte Fertigung.

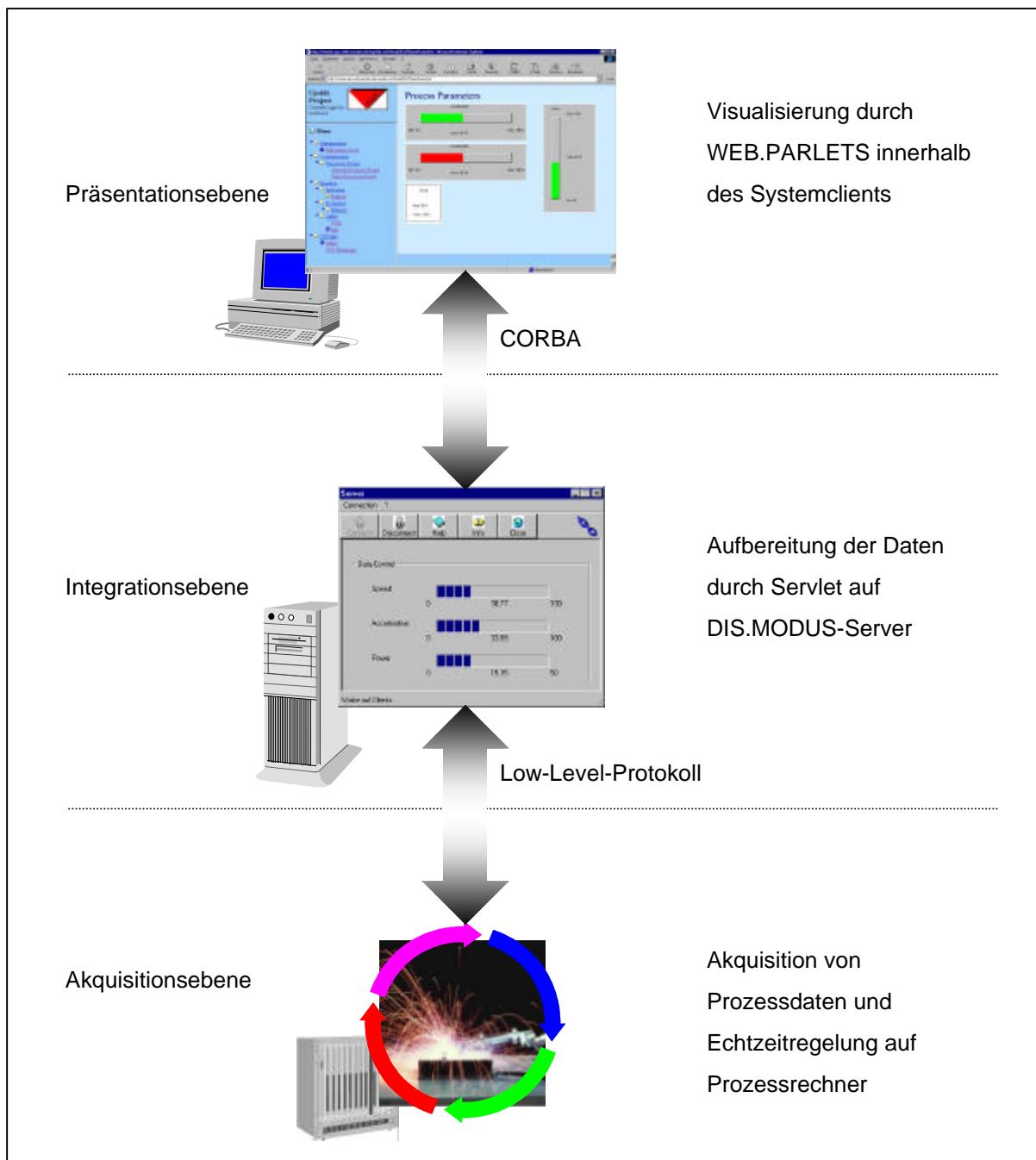


Abbildung 6-10: Ebenenmodell der verteilten Visualisierung und Regelung von Prozessparametern

Die hohe Anforderungsdivergenz erfordert den Einsatz objektorientierter Middleware in Kombination mit Intranet-Technologie und der integrierten Echtzeitplattform. Das System muss daher auf Basis des Konzeptes aus Kapitel 5.2.1 auf drei verschiedenen Ebenen arbeiten, für die spezialisierte Teilsysteme entwickelt wurden. Abbildung 6-10 gibt einen Überblick über die hier eingesetzte Technik. Man erkennt hier deutlich die verschiedenen Ebenen, auf denen die Teilmodule ausgeführt werden.

Auf der Seite der Echtzeitplattform wurde dazu ein Modul entwickelt, dass Daten z.B. von Sensoren oder der Schweißquelle empfängt und zur weiteren Verarbeitung aufbereitet. In dieses Modul können echtzeitfähige Regelalgorithmen integriert werden, die eine Optimierungsfunktion für den Prozess während der Laufzeit übernehmen.

In der exemplarischen Realisierung kommuniziert dieses Echtzeitmodul über ein einfaches, aber leistungsfähiges Protokoll, das dem Protokoll des Online-Simulationssystems WEB.MOVE angepasst ist, mit einem Servlet, das in der Umgebung des Servers des Unterstützungssystems abläuft. Das Servlet ist vollständig in Java programmiert und verwendet in Richtung des Clients des Unterstützungssystems den Middlewarestandard CORBA [CUR97].

Mithilfe der Middleware ist es möglich, nicht nur die Prozessparameter selbst, sondern gleichzeitig auch Methoden zur Visualisierung der Parameter zum Client zu übertragen.

In Anlehnung an die Begriffe Servlet bzw. Applet wurde die Bezeichnung WEB.PARLET (**Web-based Parameter Applet**) für die clientseitigen Module eingeführt. Durch verschiedene Systemparameter innerhalb der das WEB.PARLET einbettenden HTML-Seite können die folgenden Konfigurationen vorgenommen werden:

- ➔ Balkenanzeige vertikal, horizontal oder deaktiviert
- ➔ Farbe und Größe der Anzeige
- ➔ Bezeichnung des Parameters
- ➔ Skalierung der Balkenanzeige
- ➔ Maximal- und Minimalwerte zum Auslösen von Aktionen bzw. Warnmeldungen

Die Visualisierung von Prozessparametern ist somit flexibel einsetzbar. Für nicht-echtzeitkritische Aufgaben kann sie beim unter- bzw. überschreiten von Minimal- und Maximalwerten ein binäres Signal generieren. Das Signal kann entweder in Form eines Stops an die Echtzeitplattform übertragen oder in Form einer optischen oder akustischen Warnung am Client ausgegeben werden. Echtzeitkritische Parameter

können mithilfe von lokalen, in die Echtzeitplattform integrierten Regelungsalgorithmen überwacht werden.

Das System der WEB.PARLETS soll an andere Protokolle und Prozesse adaptiert werden können und zu diesem Zweck über entsprechende Schnittstellen verfügen. Im Zuge der Entwicklungen sollte daher an einem Implementierungsbeispiel die Flexibilität der Plattform nachgewiesen werden. Im Bereich der Neuentwicklungen von Maschinensteuerungen tendieren heute die Hersteller mehr und mehr dazu, standardisierte Lösungen anzubieten. Man verspricht sich von dieser Entwicklung eine Kostenersparnis, die mit einem potenziellen Marktvorteil, der durch ein offenes Steuerungskonzept erreicht werden soll, einhergeht.

Die internationale Forschungsinitiative OSACA (Open System Architecture for Controls within Automation Systems) hat aus diesem Grund eine Softwareplattform entwickelt, die Maschinensteuerungen mit einem offenen Interface ausrüstet und somit die Kommunikation mit anderen Steuerungen, Sensoren und Aktuatoren herstellerunabhängig ermöglichen soll [OSA97], [OSI98], [OSP98].

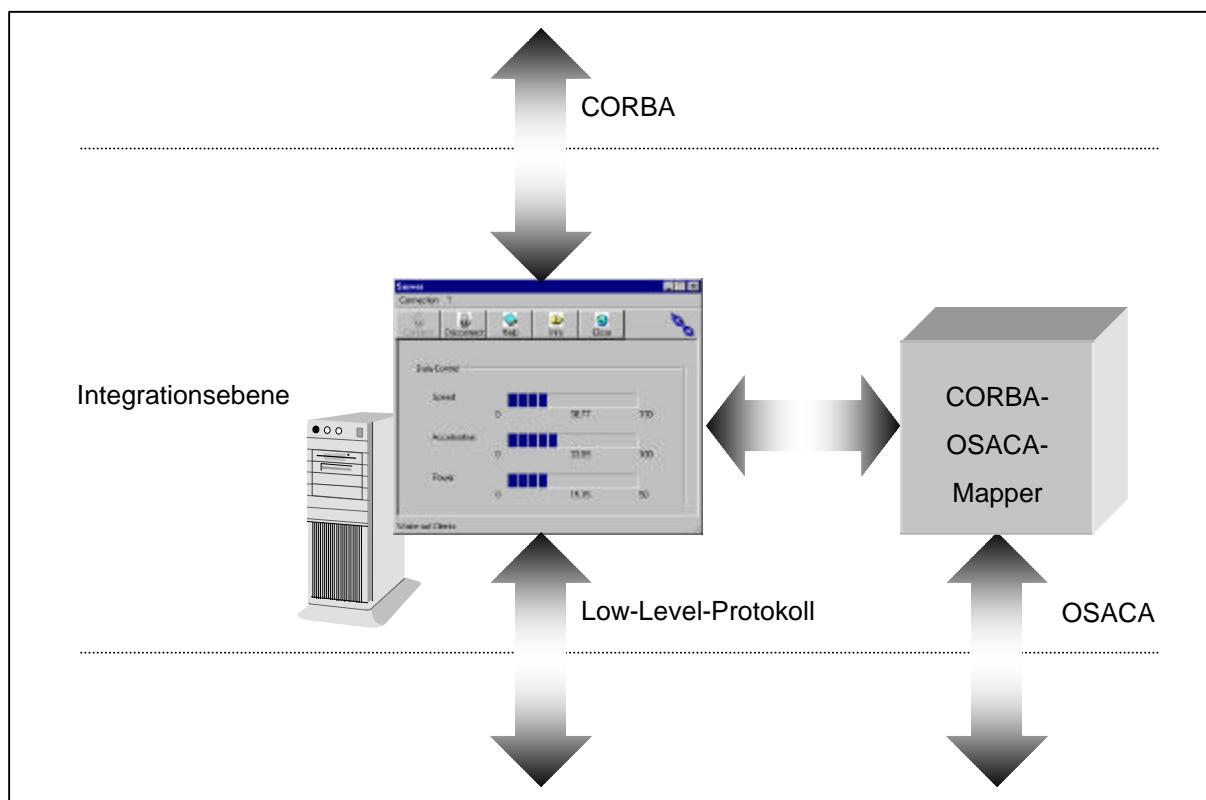


Abbildung 6-11: Integrationsebene mit integriertem CORBA-OSACA-Mapper

Durch die Entwicklung eines prototypischen Interfaces zwischen dem WEB.PARLETS-Servlet und einem OSACA-Netzknoten konnten Flexibilität und Adaptivität der im Unterstützungssystem gewählten Plattform demonstriert werden.

Dabei bildet der sogenannte CORBA-OSACA-Mapper die Schnittstelle zwischen der OSACA-basierten Steuerung und dem CORBA-basierten Servlet (Abbildung 6-11). Der OSACA-CORBA-Mapper ermöglicht somit die Visualisierung von Prozessparametern einer OSACA-konformen Maschinensteuerung mithilfe der WEB.PARLET-Technik. Der Mapper akquiriert dabei die Prozessdaten über das OSACA-Protokoll aus dem Fertigungsprozess, extrahiert aus diesem die gewünschten Parameter und überträgt diese direkt zum lokalen Servlet. Das Servlet seinerseits stellt den im Client des Unterstützungssystems ablaufenden WEB.PARLETs die Daten zur Visualisierung zur Verfügung.

6.3.2 Interaktive Simulation und Visualisierung

Die Fertigung von komplexen Bauteilen im Schiffbau stellt hohe Ansprüche an das Verständnis des Fertigungsprozesses und des Zusammenspiels der Teilprodukte. Die meist komplexen Geometrien der Bauteile sind ohne anschauliche Skizzen oder Modelle nur schwierig zu verstehen und stellen selbst erfahrene Werker oft vor schwierige Aufgaben. Bei der Fertigung von komplexen Bauteilen treten aufgrund dieser Tatsache immer wieder Fehler auf, die mithilfe von auf den Anwenderkreis zugeschnittenen Softwarelösungen vermieden werden könnten.

Im Rahmen des benutzerorientierten Unterstützungssystems wurden daher verschiedene Konzepte verfolgt, die in der Realisierung und späteren Integration in das Gesamtkonzept mündeten.

In einem ersten Schritt wurde untersucht, welche Anforderungen für die Visualisierung des Werkstücks sowohl aus der Perspektive des Anwenders als auch aus technischer Sicht existieren. Basierend auf diesem Anforderungskatalog wurde die Realisierbarkeit bewertet und die Konzepte daran adaptiert.

Neben der Visualisierung des Werkstücks ist der Komplexitätsgrad des zu fertigenden Werkstücks maßgebend für die Komplexität des Fertigungsprozesses. Erschwerend kommt hinzu, dass viele der Fertigungsprozesse im Schiffbau teilautomatisiert sind. Meist werden nur sehr geringe Losgrößen der Teilprodukte in der Arbeitszelle angefertigt, sodass jede Aufgabe die Neukonfiguration der Zelle sowie die Erstellung bzw. Anpassung eines NC- oder Roboterprogramms erfordert.

6.3.2.1 Zielproduktvisualisierung

Die Visualisierung von Werkstücken soll das zu fertigende Produkt oder Teilprodukt auf anschauliche, möglichst realistische Weise darstellen. Mit Hilfe von 3D-Technologien soll hier dem Werker eine einfach zu bedienende Lösung angeboten werden. Wichtiger Aspekt für die Nutzbarkeit dieser Anwendung ist die Möglichkeit, komplexe Bauteildaten aus CAD-Datenbeständen so zu visualisieren, dass ihre

Präsentation den Werker nicht verwirrt, sondern ihm eine effektive Hilfestellung bietet.

Aufgrund der verschiedenen 3D-Datenformate sowie der durch die verschiedenen Werkstücktypen und Nutzer bedingten Anforderungen sollen in die Implementierungen des Unterstützungssystems der beiden Werkstätten unterschiedliche Visualisierungssysteme integriert werden. Die Integration von werkstattspezifischen Systemen soll gleichzeitig der Verifikation der Integrationsfähigkeit und der Modularität des Benutzerunterstützungssystems dienen.

Werkstückvisualisierung in der Rohrwerkstatt

Die erste Implementierung eines Visualisierungsmoduls wurde in der Rohrwerkstatt realisiert. Das Visualisierungsmodul soll die in Abbildung 2-6 dargestellte Skizze des zu fertigenden Rohres, die häufig Ursache für Fertigungsfehler war, ersetzen. Die häufigsten Fehler traten bei der Einschätzung der Winkel der Rohr-Teilstücke sowie bei der Stellung der Teilstücke untereinander auf.

Das hier eingesetzte System wurde in das Produktdatenmodell der Schiffswerft integriert und verfügt so stets über aktuelle Datensätze. Änderungen der Planungsabteilung können somit innerhalb kürzester Zeit in die Produktion überführt werden. Verbesserungsvorschläge oder lokalisierte Fehler in der Planung können somit sofort über das allgemeine Produktdatenmodell in die Fertigung weitergegeben werden. Das hiermit mögliche schnelle Feedback der Planungsabteilung auf Anfragen oder Änderungsvorschläge der Werker stellt eine neue Motivation für die Verbesserung der Produkte dar.

Aufgrund des in der Rohrwerkstatt verwendeten CAD-Datenformats muss hier ein spezieller Visualisierungsclient eingesetzt werden. Es bot sich die Integration des sogenannten „GeoOutlook“-Systems der Firma Bentley an.

Dieser Client ist in der Lage, das Format der Produktdaten objektbezogen auszulesen bzw. zu durchsuchen. Dabei verbindet sich GeoOutlook über das Netzwerk mit der Produktmodell-Datenbank der Konstruktionsabteilung. Durch die Eingabe der Serienbezeichnung des Schiffes, der laufenden Nummer des Schiffes und der Bezeichnung des Rohres wird von GeoOutlook eine ODBC-konforme Abfrage der Produktmodell-Datenbank generiert. Wird das entsprechende Werkstück gefunden, wird das Modell an GeoOutlook übertragen.

Nach dem Ladevorgang erscheint die Darstellung des Rohres in einer schattierten 3D-Ansicht. Der Anwender kann die Ansicht mithilfe der Maus oder der Tastatur drehen, vergrößern oder automatisch an das Darstellungsfenster anpassen (Abbildung 6-12).

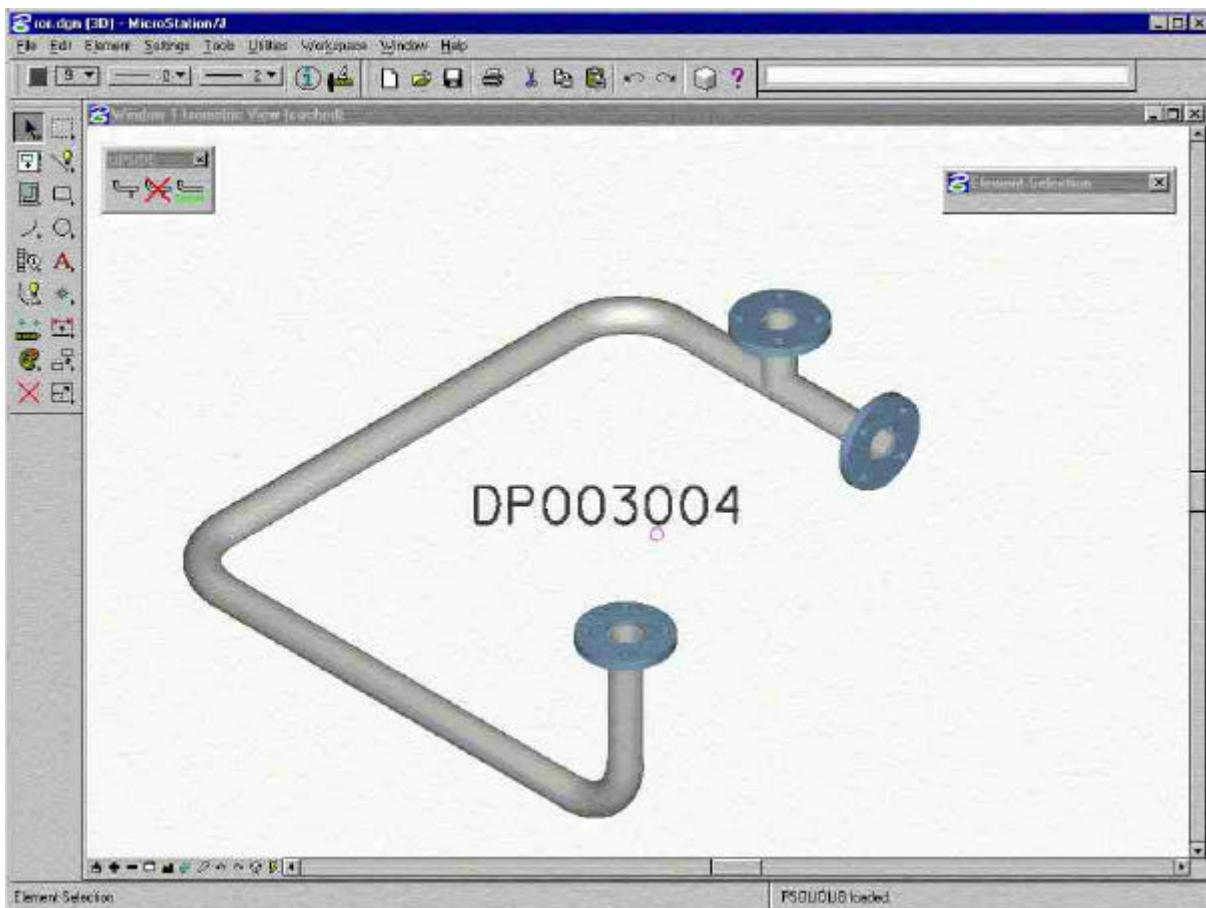


Abbildung 6-12: Werkstückvisualisierung mit GeoOutlook

Die Integration der als eigenständiges Fenster ablaufenden Applikation „GeoOutlook“ in das Unterstützungssystem wurde mithilfe der WEB.LAP-Technik realisiert, die in Kapitel 6.2.2 detailliert beschrieben wurde.

VRML-Viewer

Die Montagewerkstatt erfordert die Integration eines anderen 3D-Visualisierungssystems, da einerseits die Erfahrung der Werker mit PC-Systemen, wie sie in der Rohrwerkstatt vorzufinden ist, nur ansatzweise vorhanden ist und andererseits hier die Daten der Produkte in einem proprietären, VRML sehr ähnlichen Format vorliegen. Hier bietet sich der Einsatz der VRML-Technologie an.

Durch ihren Einsatz wird das System der Forderung nach einfacher Bedienbarkeit auch durch ungeübte Benutzer gerecht. Durch seinen hohen Interaktivitätsgrad – der Anwender kann mithilfe der Maus alle Operationen intuitiv ausüben – ist es besonders leicht erlernbar. Die wegen der PlugIn-Technologie hervorragende Integration in die Bedienoberfläche des Unterstützungssystems lässt diese Lösung als besonders anwenderfreundlich erscheinen.

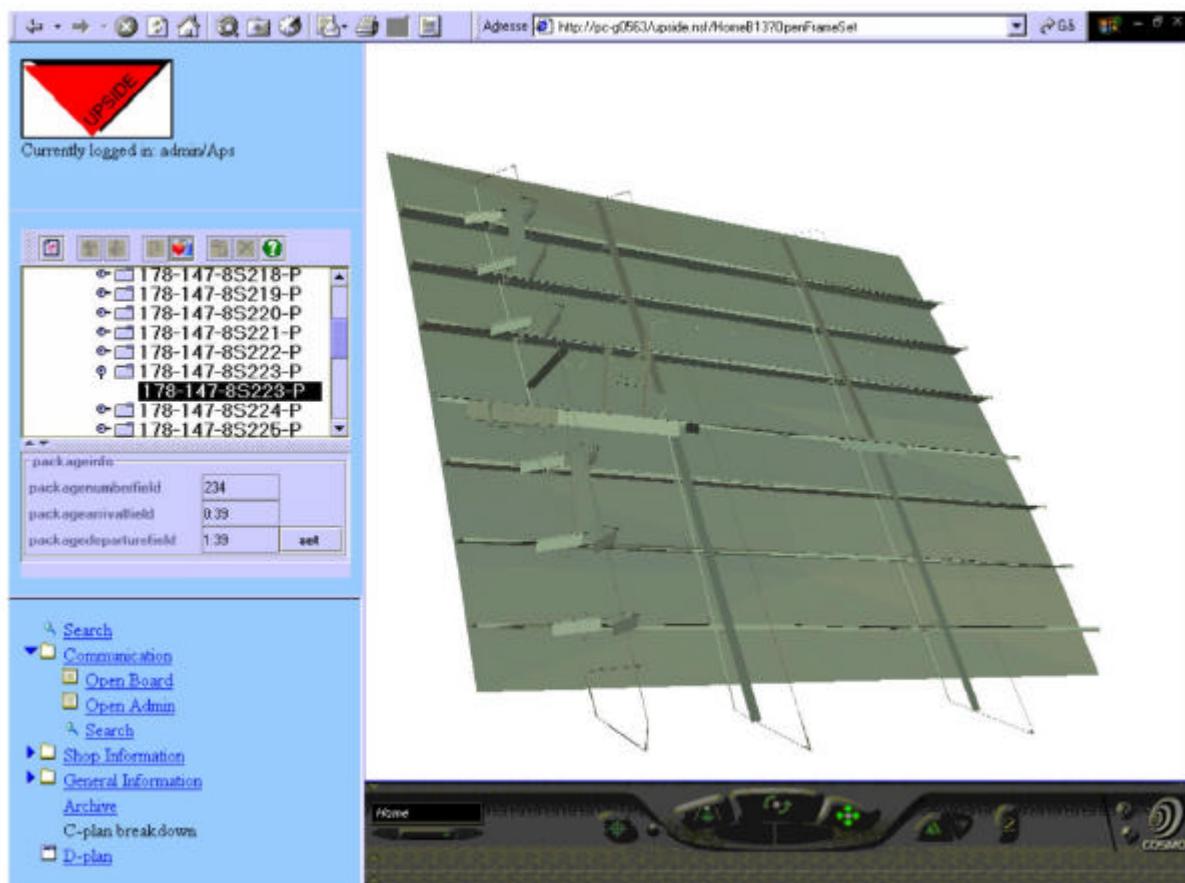


Abbildung 6-13: VRML-Werkstückvisualisierung

Basierend auf einer Liste der Aufträge im „LineController“ generiert das System Zusatzinformationen, unter anderem auch die entsprechenden VRML-Modelle der Werkstücke, die auf dem Server abgelegt werden.

Datenbasis der hier generierten VRML-Modelle ist die Datenbank des für diese Werkstücke eingesetzten CAD-Systems, die über eine ODBC/SQL-Verbindung vom Server des Unterstützungssystems abgefragt wird. Das von der CAD-Datenbank zurückgelieferte Werkstückmodell wird anschließend konvertiert und als VRML-Datei auf dem Server gespeichert.

Wird nun im LineController die Visualisierungsfunktion aufgerufen, generiert das System eine Verknüpfung, die auf die dem markierten Auftrag zugehörige Datei mit dem VRML-Modell verweist. Diese Verknüpfung wird dann dem rechten Frame des Browsers übergeben, der das Werkstückmodell lädt und darstellt (Abbildung 6-13). Das System verwendet zur Visualisierung das Web-Browser-PlugIn „CosmoPlayer“ (Cosmo Software). Das PlugIn unterstützt das Heranzoomen, Drehen und Verschieben des Modells mithilfe der Maus. Die Darstellung kann als Drahtmodell oder schattiert erfolgen.

6.3.2.2 Verifikation durch direkte Prozesskopplung und virtuelle Steuerungskomponenten

Durch die direkte Prozesskopplung des Verifikationssystems für die vollständig oder teilweise automatisierte Fertigung im Bereich des Schweißens mit Robotern sollen dem Anwender neue Interaktionsoptionen sowie größere Gestaltungsspielräume bei der Erstellung von Programmen zur Verfügung gestellt werden. Ziel dieses Systems ist es, dem Werker vor, während und nach dem Prozess nicht nur ein Simulationswerkzeug, sondern ein in das benutzerorientierte Unterstützungssystem sowohl technisch als auch inhaltlich eingebettetes, vielfältig nutzbares Verifikationswerkzeug anzubieten.

In den folgenden Abschnitten wird zunächst die technische Seite der Realisierung geschildert. Im Anschluss daran werden die anwendungsorientierten Aspekte aufgezeigt.

Bei der Realisation des **web-basierten multimodalen Verifikationswerkzeugs**, im Folgenden kurz als WEB.MOVE bezeichnet, wurde nach objektorientierten Entwurfsmethoden vorgegangen. Komplexe kinematische Objekte können so in hierarchisch gegliederte Bäume geordnet werden.

Auf das Geometriemodell wurde auf Basis des Konzeptes von [BEN99] eine leistungsfähige Kollisionskontrolle aufgesetzt. Verbindendes Modul zwischen Geometriemodell und Kollisionskontrolle ist das Kinematikmodell.

Geometriemodell

Zur Visualisierung und echtzeitfähiger Simulation des Bewegungsablaufs von automatisierten Maschinen wird ein Geometriemodell verwendet, dass den Forderungen nach Benutzerfreundlichkeit und gleichzeitiger Leistungsfähigkeit genügt. Zusätzlich zur reinen Simulation wurden hier Interaktionsoptionen geschaffen, die sowohl dem Anwender als auch autonomen Überwachungsmechanismen, sogenannten Watchdogs, während der Prozesslaufzeit Eingriffe erlauben.

Die Visualisierung baut dabei auf ein Geometrie- und Kinematikmodell auf. Diese Modelle werden unter Verwendung des OpenGL-Standards auf den virtuellen Betrachterstandpunkt projiziert. Daraus ergibt sich ein Gesamtbild, das den realen Prozessablauf relativ genau abbildet. Zu diesem Zweck werden die Geometrien mithilfe der Festkörpermodellierung auf mathematische Körper abgebildet, die die Grundlage für alle weiteren Berechnungen liefern.

Nach [BEN99] bestehen die Vorteile der Festkörpermodellierung in

- ➔ einer nutzergerechten Darstellungsform,
- ➔ der Modellierbarkeit von Freiformflächen,
- ➔ der vereinfachten Erstellung von Explosionsdarstellungen,
- ➔ der geometrischen Datenkonsistenz sowie
- ➔ einer vergleichsweise schnellen Prototypenerstellung.

Die Grafikbibliothek stellt eine Vielzahl von Diensten für die Projektion der Geometriemodelle zur Verfügung. So können vergleichsweise einfach z.B. Betrachterposition, Lichteinfluss oder Skalierung an die jeweilige Situation angepasst werden. Neben der Konstruktion von Geometrien auf Polygon- bzw. Begrenzungsf lächenbasis (B-REP, Boundary Representation) bieten OpenGL-Hilfsbibliotheken die folgenden Hilfskörper an, die durch Zusammenfügen komplexere Körper ergeben: Kugel, Zylinder, Konus, Quader, Würfel, Torus, Ikosahedron, Oktahedron, Tetrahedron und Dodekahedron. Die sich daraus ergebende Kette von sogenannten „Primitiven“ wird auch als CSG-Baum (Constructive Solid Geometry) bezeichnet.

Da die später beschriebene Kollisionserkennung auf die B-REP-Repräsentationsform angewiesen ist, wird das Geometriemodell nach diesem Verfahren erstellt. Wegen der erforderlichen Performance für die echtzeitnahe Prozess-Simulation musste ein Verfahren eingeführt werden, dass die Komplexität der Körper reduziert und somit eine schnellere grafische Projektion der Modelle erlaubt. Dabei muss beachtet werden, dass der Körper für die sinnvolle Überprüfung auf mögliche Kollisionen nicht verkleinert wird. Durch den Einsatz des von Barber, Dobkin und Hihdanpaa [BDH96] entwickelten Hüllverfahrens „Quickhull“ war eine erhebliche Steigerung der Performance unter nur geringfügigen Einbußen der Modelloptik und -konsistenz möglich. Das Hüllverfahren legt dabei eine konvexe Hülle um einen komplexen Körper, dessen Polygonanzahl sich unter Umständen dadurch drastisch reduziert (Abbildung 6-14).

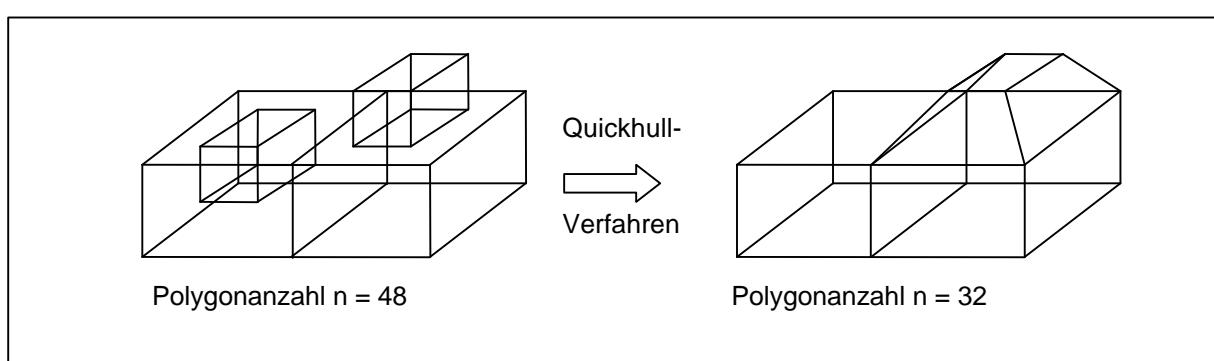


Abbildung 6-14: Polygonreduktion durch Quickhull-Verfahren

Nachteil des Verfahrens ist, dass konkave Körper geschlossen werden.

Das Beispiel in Abbildung 6-14 demonstriert die deutliche Reduktion der zu berechnenden Polygone.

Kinematikmodell

Basierend auf dem oben beschriebenen

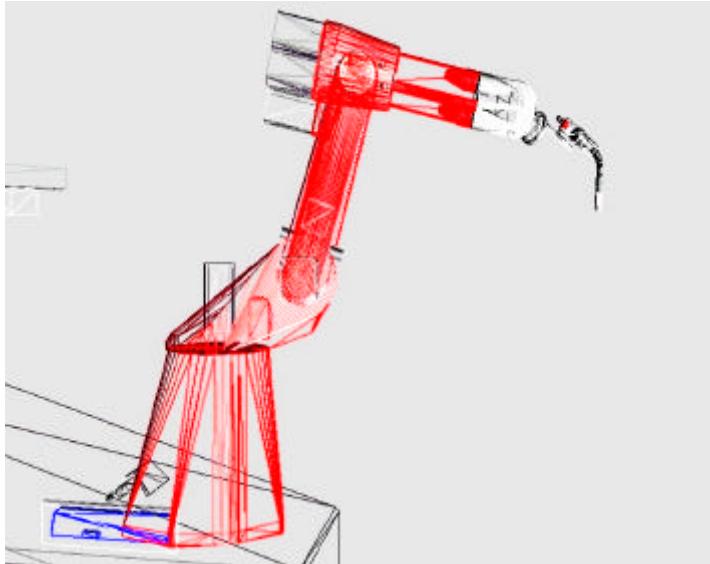


Abbildung 6-15: Geometrie- und Kinematikmodell des Reis-Roboters

Geometriemodell wurde ein Kinematikmodell entwickelt, dass die grafische Simulation und gleichzeitig die semantische Verifikation des Fertigungsprozesses ermöglicht. Das Kinematikmodell nutzt die in Kapitel 5.2.2 beschriebenen Verfahren, um die Bewegung von redundanten Kinematiken mit bis zu 255 Achsen mathematisch zu beschreiben. In der hier realisierten Lösung kommt der Reis-Roboter (Abbildung 6-15) mit sechs

Achsen zur Anwendung. Optional lassen sich die beiden Zusatzachsen des Positioniertisches mit einkoppeln.

Kollisionskontrolle

Die integrierte Kollisionserkennung ist eines der wesentlichen Elemente der der Visualisierungseinheit hinterlegten funktionalen Simulation. Das hier implementierte Kollisionsverfahren basiert auf dem von [BEN99] entwickelten System. Dieses Verfahren vereint die hierarchische Kollisionskontrolle mit der sogenannten exakten Methode zur Kollisionserkennung. Das hierarchische Verfahren nach [HEN92] verwendet Kompositionen von Modellelementen und prüft in einem ersten Schritt auf eine mögliche Kollision. Kollidieren die Kompositionen nicht, kann daraus gefolgt werden, dass auch die Einzelkomponenten der Komposition nicht kollidieren.

Die Prüfung des so definierten Baums ergibt eine Kollisionsmatrix, die spezifiziert, welche Kompositionen eine Kollision beinhalten. Tabelle 6-3 zeigt eine exemplarische Kollisionsmatrix für einen sechsachsigen Roboter. In diesem Beispiel wurde eine Kollision des Teilmodells K_{56} des Roboters mit dem Teilmodell K_{12} identifiziert.

<i>Teilmodelle des Roboters</i>	K_{01}	K_{12}	K_{23}	K_{34}	K_{45}	K_{56}	K_{6T}
K_{01}	1	0	0	0	0	0	0
K_{12}	0	1	0	0	0	1	0
K_{23}	0	0	1	0	0	0	0
K_{34}	0	0	0	1	0	0	0
K_{45}	0	0	0	0	1	0	0
K_{56}	0	1	0	0	0	1	0
K_{6T}	0	0	0	0	0	0	1

Tabelle 6-3: Exemplarische Kollisionsmatrix eines sechsachsenigen Roboters

Man erkennt hier die Gesetzmäßigkeiten, die eine Reduktion der zu untersuchenden Kompositionen des Roboters ermöglichen:

- Eine Komposition kann logisch nicht mit sich selber kollidieren, wenn sie keine kinematischen Elemente beinhaltet. Kompositionen sind daher so zu definieren, dass sie in sich geschlossene statische Objekte sind. Eine mathematische Prüfung einer Komposition mit sich selber ergibt stets eine Kollision.
- Zwei Kompositionen müssen nur in einer Richtung auf Kollision geprüft werden. Kollidiert A mit B, so gilt im Umkehrschluss, dass B auch mit A kollidiert.

Aus den obigen Gesetzmäßigkeiten folgt, dass eine Kollisionsmatrix immer spiegelsymmetrisch zur Diagonalen ist und dass die Elemente der Diagonalen stets zu eins gesetzt werden können. Die daraus folgende Redundanz der Kollisionsmatrix ist in Tabelle 6-3 grau hinterlegt.

Durch Beachtung dieser einfachen Regeln konnte die Anzahl der Prüfungen nach der hierarchischen Methode deutlich reduziert werden. Ist n_p die Zahl der zu prüfenden Kollisionen vor der Reduktion und n_k die Zahl der zu prüfenden Elemente der Werkzelle, so gilt

$$n_p = n_k^2 \quad (6-5).$$

Für die Zahl der zu prüfenden Kollisionen nach der Reduktion n'_p ergibt sich

$$n'_p = \frac{n_k}{2} (n_k - 1) \quad (6-6).$$

Die Leistung des Algorithmus verhält sich linear zur Zahl der durchzuführenden Kollisionsprüfungen. Damit ergibt sich durch die Reduktion eine Leistungssteigerung von

$$\frac{n_p}{n'_p} = \frac{2n_k}{n_k - 1} \quad (6-7).$$

Der Leistungsgewinn sinkt also bei höherer Kompositionszahl, beträgt jedoch minimal 100% (Abbildung 6-16).

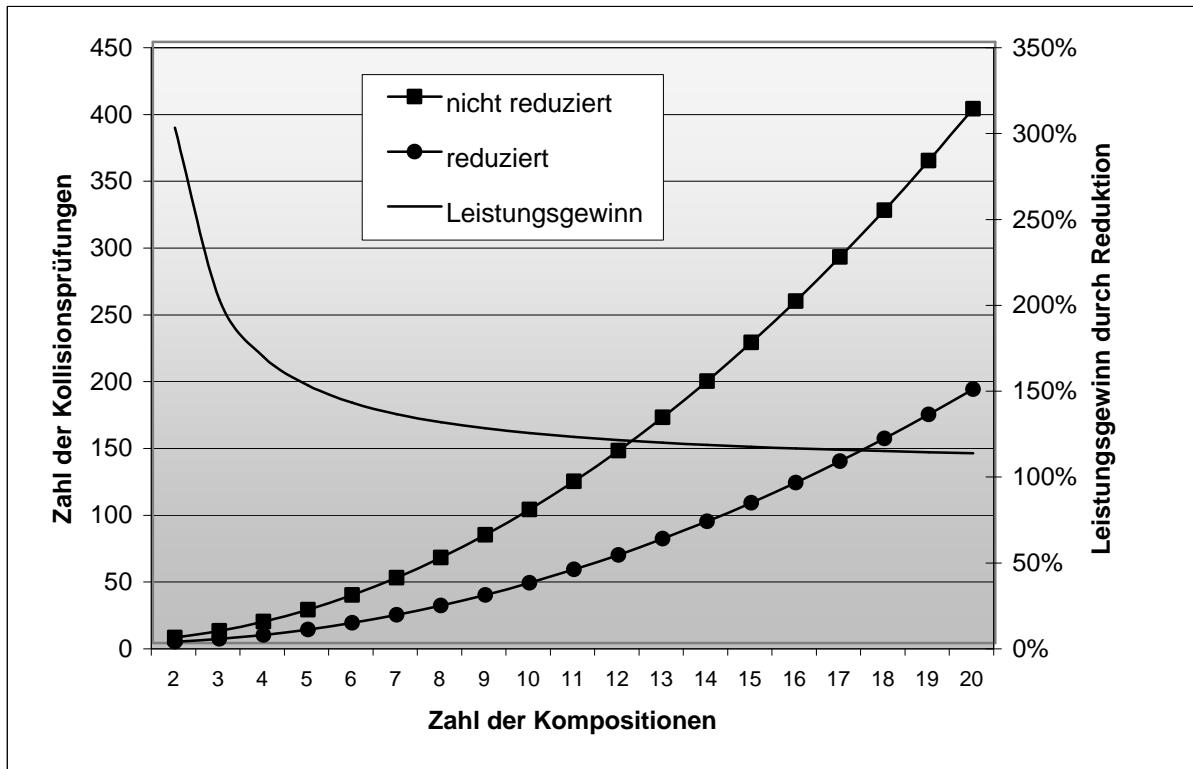


Abbildung 6-16: Leistungsgewinn der Kollisionserkennung durch Reduktion der Kompositionskombinationen

Die Methode der exakten Kollisionskontrolle dient zur genaueren Eingrenzung der Kollision innerhalb der Kompositionen. Es verwendet sogenannte Bounding-Boxen, die ein geometrisches Element umschließen, um festzustellen, ob das jeweilige Element möglicherweise ein anderes schneidet. Ist dies der Fall, wird mithilfe einer exakten geometrischen Berechnung geprüft, ob sich das Element, z.B. ein konvexer Polyeder, tatsächlich mit einem anderen schneidet. Die geometrische Berechnung nach [LIC91] liefert neben der binären Kollisionsinformation gleichzeitig die Abstände zwischen den geometrischen Elementen. Durch die Definition von Toleranzbereichen für die Abstände lassen sich so im Werkstatteinsatz sehr hilfreiche Warnmeldungen beim Unterschreiten eines bestimmten Abstands generieren. Positionen mit hohem Kollisionspotenzial bei der Programmierung des Roboters können so bereits bei einem Testlauf effektiv vermieden werden.

Das Kollisionsmodul kommuniziert direkt mit der Echtzeiteinheit der Werkzelle und ist somit in der Lage, im Falle des Erreichens einer kritischen Position den sofortigen

Stopp der Roboterbewegung auszulösen. Die Stoppmeldung versetzt den Roboter in einen Pausenzustand, in dem er auf mögliche Benutzerinteraktionen, z.B. auf die manuelle Korrektur der Bahn, wartet. Im Anschluss an die Korrektur kann der Programmablauf an der nächsten Position fortgesetzt werden.

Implementierung des Moduls WEB.MOVE

Das interaktive Visualisierungs- und Simulationswerkzeug WEB.MOVE soll den Werker in die Lage versetzen, den Fertigungsprozess vor und während des Ablaufs zu verifizieren. Gleichzeitig stellt es eine Plattform für die autonome und für den Anwender transparent ablaufende automatische Prozessüberwachung dar.

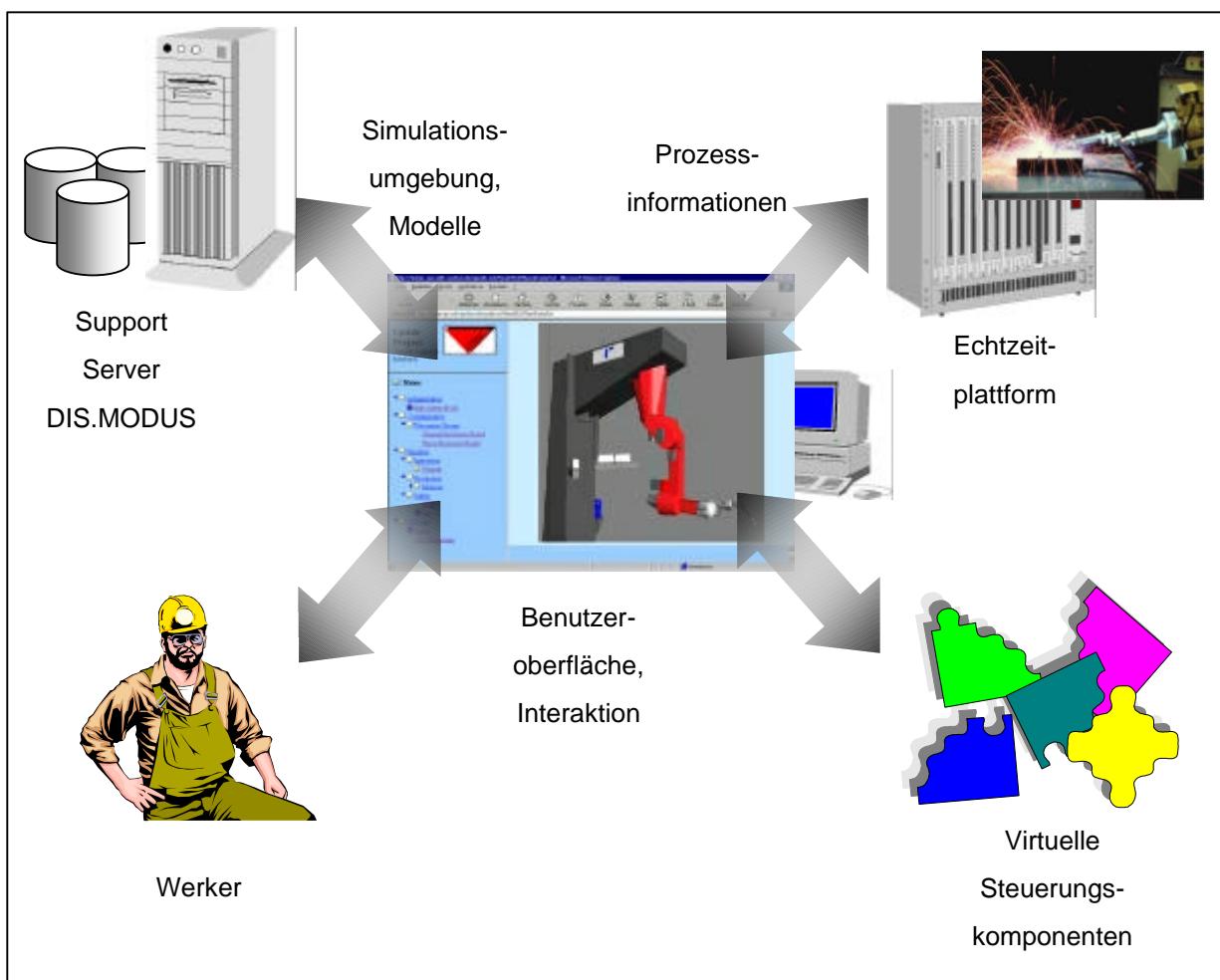


Abbildung 6-17: Schematische Struktur des Visualisierungs- und Simulationswerkzeugs

WEB.MOVE wurde vollständig in das Unterstützungssystem DIS.MODUS integriert und erweitert somit nahtlos dessen Funktionalität. Voraussetzung für die nahtlose Integration ist die Implementierung als intranetfähige, browserkonforme Anwendung. In Abbildung 6-17 ist schematisch die implementierte Plattform dargestellt.

Vom DIS.MODUS-Server wird im ersten Schritt die dynamisch generierte HTML-Umgebung in den Browser auf Werkstattebene übertragen. Diese HTML-Umgebung enthält alle für die Ausführung von WEB.MOVE notwendigen Informationen wie den Typ und Modellbezeichnung der zu visualisierenden Maschine, in dieser Werkzelle unterstützte Interaktionen sowie die prozessgebundenen technischen Parameter. Der Web-Browser lädt anschließend das hier installierte TCL/Tk-Runtime Plugin, das vom Support Server das Simulationsskript sowie das Geometrie- und Kinematikmodell anfordert.

Nach einer Initialisierungsphase stellt das Simulationswerkzeug eine Verbindung zu der Echtzeitplattform der vom Support Server identifizierten Maschine her. Ist die Verbindung aktiv, wird der Status der Maschine geprüft und analysiert, welche Interaktionsfunktionen im gegenwärtigen Status angeboten werden können und entsprechend gestartet werden sollen. Alternativ kann das System eine Verbindung zu virtuellen Steuerungskomponenten herstellen, die das Verhalten der Steuerung sowie den Fertigungsprozess modellbasiert nachbilden.

Verifikation durch direkte Prozesskopplung

Die Multimodalität des Verifikationswerkzeugs basiert auf den verschiedenen Betriebszuständen, die dem Anwender zur Verfügung stehen.

Einerseits kann WEB.MOVE zur Verifikation durch direkte Prozesskopplung verwendet werden. Dieser Betriebszustand wird zur Analyse des Prozesses während seiner Laufzeit verwendet und ermöglicht so die direkte Interaktion des Anwenders bzw. der autonomen Verifikationsfunktionen mit dem Fertigungsprozess. Basis dieses Online-Betriebszustands ist die bereits beschriebene Echtzeitplattform in Kombination mit dem in Kapitel 6.3.1 spezifizierten Protokoll für die echtzeitnahe Kopplung des Fertigungsprozesses unter Verwendung von verbreiteten Netzwerkstandards.

Pre-Prozess-Verifikation durch virtuelle Steuerungskomponenten

Andererseits dient das Visualisierungs- und Simulationssystem zur Verifikation eines Prozesses vor seiner Laufzeit (Pre-Prozess-Verifikation). Hierbei greift WEB.MOVE nicht auf die Echtzeitplattform und damit auf reale Fertigungseinrichtungen zurück, sondern verwendet als Datenquelle virtuelle Steuerungskomponenten, die eine möglichst realitätsnahe Emulation der realen Fertigung darstellen.

So wurde exemplarisch eine virtuelle Robotersteuerung realisiert, die die Off-line-Simulation von Programmen und Prozessen ermöglicht. Basis für die Prozessemulation waren die in Analogie zur realen Robotersteuerung

implementierten Algorithmen. Diese Algorithmen verwenden das bereits beschriebene Kinematikmodell des Roboters.

Die virtuellen Steuerungskomponenten sind nicht an eine bestimmte Plattform gebunden, sondern können – je nach Anforderung – an verschiedene Plattformen angepasst werden. Beispielsweise stellt die Ausführung auf der Echtzeitplattform der Werkzelle zwar den Idealfall der Simulation dar, da hier auch der Interpolationstakt des Roboters exakt nachgebildet werden kann, jedoch ist diese Plattform unter Umständen in der Werkstatt nicht jederzeit verfügbar, sodass auf andere Plattformen ausgewichen werden muss. Als sekundäre Plattform bietet sich der Client des Unterstützungssystems selbst an, da die Komponenten aufgrund des geforderten Zeitverhaltens sehr prozessorintensiv sind.

Durch den Einsatz von virtuellen Steuerungskomponenten ist es möglich, Roboterprogramme bereits vor dem Upload in die Robotersteuerung in einer realitätsnahen Umgebung zu testen. Die Abgrenzung zu existierenden Simulationssystemen ist in der werkergerechten Ausrichtung sowie in der technischen Nähe zum Prozess zu sehen.

6.3.3 Integration der Fertigungsplanung

Die Funktionalität des Unterstützungssystems wird durch die Integration von verschiedenen Modulen zur Planung und planungsbezogenen Information ergänzt (CAP – Computer Aided Planning).

Ziel der hier integrierten planungsbezogenen Systeme ist es, den Anwender bei der Planung und Ausführung seiner Aufträge zu unterstützen. Die Werkzeuge sind so ausgelegt, dass sie dem Anwender einerseits einen großen Handlungsspielraum lassen, ihn andererseits aber auch durch Vorschläge, durch die Definition von Randbedingungen und durch die Simulation von Ablaufplänen in Richtung einer optimalen Planung lenken.

Letztendlich wird der Anwender die Entscheidung über die Ausführung eines Plans treffen und somit seine Erfahrungen und sein Wissen in die Planung einbeziehen. Die Aufgabe der Planungswerkzeuge ist mit der eines Beraters zu vergleichen, der im Voraus alle nicht in Frage kommenden Eventualitäten ausschließt und so die Aufmerksamkeit des Anwenders auf die wesentlichen Entscheidungsoptionen konzentriert [TWW95].

Da die Planungsaufgaben stark vom Produkt, der Zahl und Art der Teilprodukte sowie der Ressourcenverfügbarkeit der Werkstatt abhängt, muss das Planungsmodul an die jeweilige Situation angepasst werden. Die Anpassung kann auf Basis eines konfigurierbaren Modells erfolgen, das mithilfe eines

Ressourcenmodells und eines Ablaufplans den Produktionsfluss der Werkstatt abbildet.

In den folgenden Absätzen werden vier Module beschrieben, die in das Unterstützungssystem DIS.MODUS integriert wurden und dessen Infrastruktur für die modulinterne und -externe Kommunikation nutzen. Weiterer Integrationspunkt ist die Verwendung der Benutzeroberfläche des Unterstützungssystems.

Alle Module verwenden das Client-Server-Prinzip, sodass ihre Architektur direkt auf die Architektur des Unterstützungssystems abgebildet werden kann und die zentrale Administrierbarkeit gewährleistet ist. Gleichzeitig kommt die Middlewareplattform CORBA zum Einsatz, die eine netzweite Verteilung der Daten und Methoden sowie die Anbindung weiterer Systeme ermöglicht. Für die Integration von Datenbanken wurde hier grundsätzlich ODBC angewendet.

6.3.3.1 Aufgliederung des Werkstattplanes

Das in der Werft eingesetzte hierarchische Planungssystem kommt auch hier zum Einsatz. Für die Planung auf Werkstattebene sind lediglich die sogenannten C- und D-Pläne relevant.

Ein C-Plan spezifiziert Arbeitspakete, die von einer bestimmten Werkstatt zu einem bestimmten Termin fertiggestellt werden müssen. Der C-Plan wurde bisher vom Werkstattleiter bzw. von Vorarbeitern weiter aufgegliedert und in Form von Aufträgen termingerecht an die Arbeitsstationen verteilt. Die Einführung eines neuen Planungsniveaus – des D-Plans – soll die Planung flexibler und damit effizienter gestalten. Der D-Plan spezifiziert detailliert, welche Ressource zu welcher Zeit an welchem Auftrag arbeitet.

Das hier vorgestellte Werkzeug dient der Aufspaltung des C-Plans in D-Plan-Einträge durch den Werker vor Ort. Der Werker kann so flexibel seine Arbeit einteilen und seine Erfahrung direkt in die Werkstattplanung einbringen. Das Werkzeug stellt dem Werker des weiteren detaillierte Informationen über die Zusammenhänge zwischen den Teilprodukten zur Verfügung, die bisher nur dem Vorarbeiter in Form von gedruckten Katalogen vorlagen.

Das Modul zur Aufgliederung des C-Plans wurde in Java entwickelt, sodass die nahtlose Integration in das Unterstützungssystem vollzogen werden konnte. Das Modul kommuniziert direkt mit der modellbasierten Fertigungsplanung und -simulation sowie mit der Planung der Produktionslinien, die weiter unten beschrieben werden.

6.3.3.2 Modellbasierte Fertigungsplanung und -simulation

Das Unterstützungssystem wurde durch die Integration eines kommerziellen, in der Werft in einem Feldversuch eingesetzten, Planungswerkzeuges ergänzt. Bei dem Werkzeug handelt es sich um das System CAPS (Computer Aided Process Scheduling) der Firma PDC (Prolog Development Center) [PDC98].

Ziel der modellbasierten Fertigungsplanung und -simulation ist die Erzeugung eines möglichst optimalen Ablaufplans für Aufträge aus C- und D-Plänen. Als Informationsbasis für das Planungswerkzeug dient ein logisches Modell, das aus den folgenden Daten zusammengesetzt wird:

- ➔ Physikalische Randbedingungen wie z.B. Fläche des Transportbandes
- ➔ Randbedingungen, die durch Ressourcen gegeben sind, z.B. Schweißmaschinen oder Schweißer
- ➔ Logische Zusammenhänge, z.B. Prozessreihenfolge

Das so erstellte Modell wird nun mit auftragsspezifischen Informationen wie

- ➔ Starttermin des Auftrags,
- ➔ Auslieferungstermin des Produkts und
- ➔ Typen und Dauer der verwendeten Ressourcen

gespeist. Das Planungswerkzeug erstellt nun automatisch einen Vorschlag für die Fertigungsplanung und zeigt diesen in einem Balkendiagramm an (Abbildung 6-18).

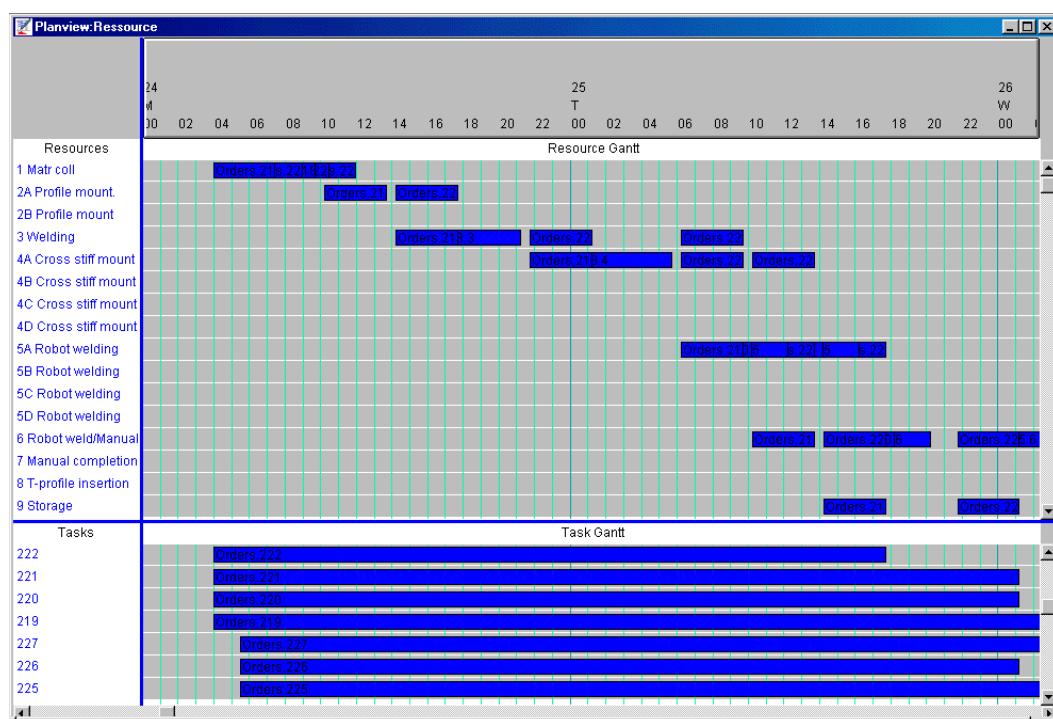


Abbildung 6-18: Balkendiagramm eines Fertigungsplans [PDC98]

Zusätzliche Ansichten des Fertigungsplans, z.B. als Flussdiagramm (Abbildung 6-19) oder als Kalender sind möglich.

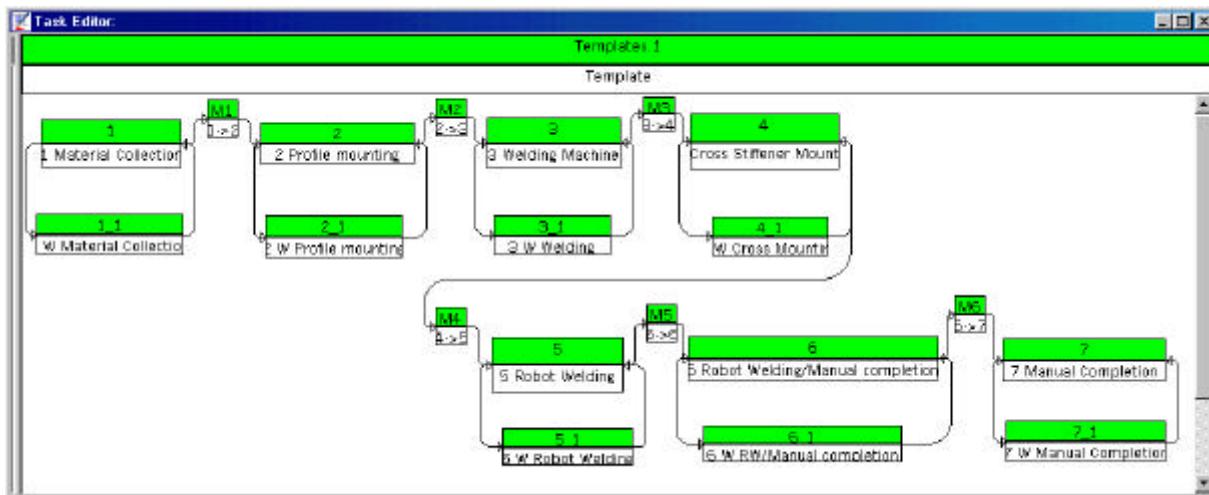


Abbildung 6-19: Flussdiagramm der Fertigung in CAPS [PDC98]

Die Darstellung des Fertigungsplans ist interaktiv, das heißt, dass der Werker mithilfe der Maus in den vorliegenden Plan eingreifen kann und somit die Generierung eines Alternativplanes initiiert. Beispielsweise kann der Werker den Plan so sehr einfach an aktuelle Gegebenheiten wie z.B. eine defekte Schweißmaschine anpassen und den Teilprozess auf einen anderen Termin verschieben. Das System generiert daraufhin einen neuen Plan, der mithilfe der integrierten Simulation verifiziert werden kann.

CAPS arbeitet nach dem Client-Server-Prinzip und konnte erfolgreich in die Architektur des Gesamtsystems eingebettet werden. Dabei dient der Server des Unterstützungssystems als Datenbankserver für CAPS. Der Client des Unterstützungssystems erlaubt mithilfe der WEB.LAP-Technik den Start des als externe Applikation ausgeführten CAPS-Clients.

6.3.3.3 Planung von Produktionslinien

Die Fertigung in der Montagewerkstatt für Strukturelemente basiert im wesentlichen auf einer Produktionslinie, in deren Verlauf die einzelnen, teilweise automatisierten Prozessschritte vollzogen werden. Das hier integrierte Werkzeug für die

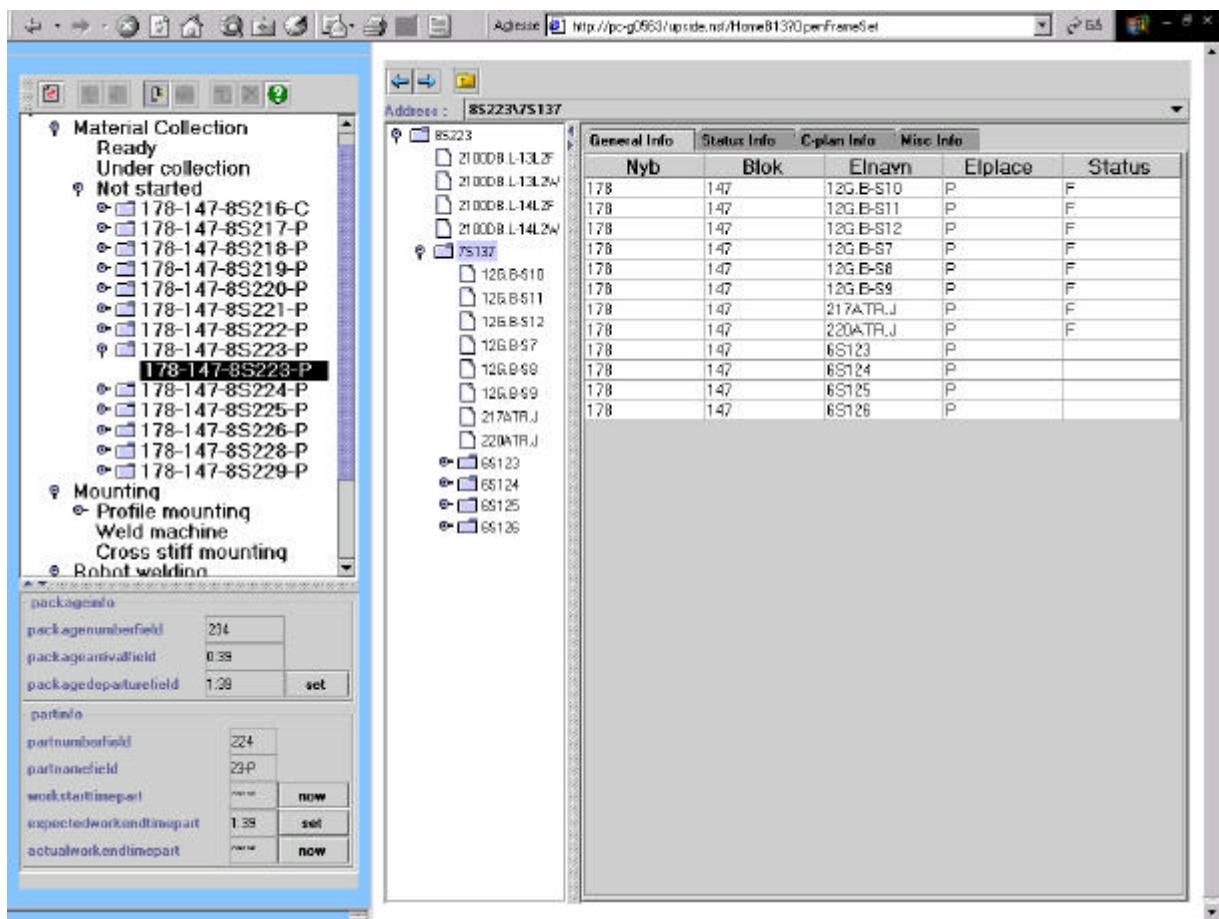


Abbildung 6-20: Überwachungswerkzeug für die Produktionslinie mit Teilproduktinformationen im Unterstützungssystem [SCH00]

Überwachung der Produktionslinie stellt die einzelnen Stationen der Linie mit jeweils drei verschiedenen Zuständen dar. Als Inhalte der Unterprozesse werden die zurzeit hier bearbeiteten Teilprodukte dargestellt (Abbildung 6-20). Ist ein solches Teilprodukt markiert, kann mithilfe der Tasten oberhalb der Baumansicht eine Übersicht des verwendeten Materials sowie eine 3D-Ansicht des Bauteils aufgerufen werden (Kapitel 0).

Das Modul lag als externe Applikation vor und konnte im Rahmen dieser Arbeit in das Unterstützungssystem integriert werden. Es kommuniziert mit dem oben beschriebenen Werkzeug zur Aufgliederung des C-Plans und mit der modellbasierten Fertigungsplanung und -simulation.

6.3.4 Prozessanalyse und Evaluierung

Die bisher beschriebenen Module dienen in erster Linie der Analyse und Regelung des Fertigungsprozesses *vor* oder *während* seiner Ausführung. Die Auswertung der Informationen konzentrierte sich dabei auf fertigungstechnische Parameter.

Zur Vervollständigung der Palette der Regelungs- und Analysewerkzeuge wurden weitere Module entwickelt, deren Aufgabe die Auswertung von Fertigungsdaten *nach Abschluss* des Prozesses ist. Der Schwerpunkt der Auswertungen liegt hier in der Verarbeitung von zeitlich gestaffelten statistischen Daten.

Es werden drei Komponenten vorgestellt, die für die Analyse und Evaluierung von abgeschlossenen Fertigungsprozessen entwickelt und in das Unterstützungssystem integriert wurden.

6.3.4.1 Manuelle Prozessevaluierung

Die erste Methode der Evaluierung bezieht sich auf Ereignisse im Umfeld der Fertigung, die nicht vorhersehbar sind oder nicht regelmäßig auftreten. Aufgrund dieser Charakteristik ist eine automatische Generierung dieser Berichte nicht möglich und es muss hier auf manuell erstellte Berichte zurückgegriffen werden.

Ein Beispiel für manuell erstellte Berichte sind textbasierte Informationen, die von einem Anwender mit entsprechenden Schreibrechten, z.B. einem Vorarbeiter, eingegeben und editiert werden können. Diese Berichte werden vom System automatisch in eine Liste übernommen, die vom Endanwender eingesehen, nach Stichworten durchsucht und nach verschiedenen Kriterien wie dem Datum oder der Überschrift sortiert werden kann. Inhalt dieser Notizen können beispielsweise Informationen über ein fertiggestelltes Gesamtprodukt, Informationen über Ergebnisse von Qualitätsuntersuchungen oder besondere Eigenschaften des zur Zeit verwendeten Rohmaterials sein.

Eine andere Anwendung von manuell erstellten Berichten wurde in der Veröffentlichung von Sicherheitsstatistiken lokalisiert. In den Werkstätten werden in regelmäßigen Abständen Statistiken über die Häufigkeit von meldepflichtigen Arbeitsunfällen erstellt und anschließend in Form von Postern in der Werkstatt veröffentlicht. Ziel dieser Veröffentlichungen ist die Sensibilisierung der Werker für unfallträchtige Tätigkeiten zur Reduktion der Unfallzahlen. Die Gestaltung der Sicherheitsberichte wurde an die bisher in den Werkstätten publizierten Poster angelehnt.

6.3.4.2 Automatisch generierte Prozessevaluierung

Neben der manuellen Integration von Berichten können Berichte auch automatisch auf Grundlage von Daten aus automatisierten Teilprozessen erstellt werden. Diese

Daten werden aufgezeichnet und so aufbereitet, dass sie übersichtlich dargestellt werden können.

In das Unterstützungssystem wurde als Beispiel für die Integration automatisch erstellter Berichte eine Roboterschweißstation in der Fertigung von Strukturelementen ausgewählt. Die Steuerung des Roboters ist über einen PC bereits mit dem Netzwerk verbunden und verfügt somit über eine wesentliche Voraussetzung für die Integration. Die Robotersteuerung ist dabei so programmiert, dass sie im Betrieb die gewünschten Daten aufzeichnet und an den PC übermittelt.

Die aufgezeichneten Daten umfassen:

- ➔ Anzahl der Roboterfehler heute
- ➔ Anzahl der Roboterfehler gestern
- ➔ Anzahl der Roboterfehler der letzten sieben Tage
- ➔ Anzahl der produzierten Teile der letzten sieben Tage
- ➔ Meter Schweißnaht pro Auftrag
- ➔ Meter Schweißnaht pro Stunde, heute
- ➔ Meter Schweißnaht pro Stunde, gestern

Abbildung 6-21 zeigt einen solchen Bericht. Die Integration der textbasierten

* * Evaluation Report * *									
Planned/Welded Meters per Job									
26 October 1999 12:31									
<hr/>									
Panel 1167-266-6d.b-p									
<hr/>									
planned meters					welded meters				
Job	vu	vd	oh	dh	total	vu	vd	oh	dh total
<hr/>									
d6dbp2_6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
d6dbp2_7	0.0	0.0	0.0	2.8	2.8	0.4	0.0	0.0	1.0 1.4
d6dbp2_8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
d6dbp3_0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
d6dbp3_1	0.0	0.0	0.0	2.8	2.8	0.0	0.0	0.0	0.0
d6dbp3_2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
d6dbp3_3	0.0	0.0	0.0	2.8	2.8	0.0	0.0	0.0	0.0
d6dbp3_4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
d6dbp3_5	0.0	0.0	0.0	2.8	2.8	0.0	0.0	0.0	0.0
d6dbp3_6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
<hr/>									
Totals:	0.0	0.0	0.0	11.2	11.2	0.4	0.0	0.0	1.0 1.4

Abbildung 6-21: Beispiel für einen automatisch generierten Bericht (Meter Schweißnaht pro Auftrag)

Informationen in das Unterstützungssystem wurde mithilfe des dynamischen Webservers erreicht. Dieser lädt bei einer Anfrage des Clients die Textdatei mit den aktuellen Informationen vom Zellenrechner und überträgt sie zum Web-Browser, wo sie dem Anwender angezeigt wird.

6.3.4.3 Statistische Evaluierung der Plankonformität

Neben der Evaluierung des Fertigungsprozesses nach technischen Gesichtspunkten wurde ein Modul entwickelt, dass eine statistische Bewertung der Plankonformität vornimmt. Als Basis der Evaluierung dient hier die Werkstattplanung, die mithilfe des vorher beschriebenen Systems CAPS realisiert wird. Das Modul wird im Folgenden kurz als WEB.PROVE (**W**ebs-based **P**roduction **V**erification and **E**valuation) bezeichnet.

Der Server des Unterstützungssystems generiert hier aus Daten der ODBC-konformen CAPS-Datenbank Vergleiche zwischen geplanten und erreichten Zielen in den folgenden Bereichen:

- ➔ Termine
- ➔ Prozesszeiten
- ➔ Durchlaufzeiten
- ➔ Ressourcenauslastung
- ➔ Produktdurchsatz

Beispielsweise ergibt sich bei einem Vergleich von geplanter und erreichter Auftragsfertigstellung eine Zeitdifferenz in Tagen, die in einem weiteren Schritt über eine größere Anzahl von Aufträgen gesehen statistisch ausgewertet werden kann. Die so gewonnenen Daten können zur Lokalisierung von kritischen Punkten in der Gesamtfertigung und damit zur Optimierung der Fertigung beitragen. Die Rückkopplung der statistischen Daten in die Planung bietet die Möglichkeit, Defizite in der Planung aufzuspüren.

Die technische Aufbereitung der Daten wurde auf der Seite des Domino-Servers durch den Einsatz von ODBC realisiert. Damit besteht die Möglichkeit, die Daten netzwerkweit zu akquirieren und anschliessend in eine anwendergerechte Form zu bringen. Zusätzlich wird die Auswertung der Daten grafisch dargestellt und mit Trendlinien versehen, um Schwerpunkte z.B. in der Auftragsverzögerung zu lokalisieren (Abbildung 6-22).

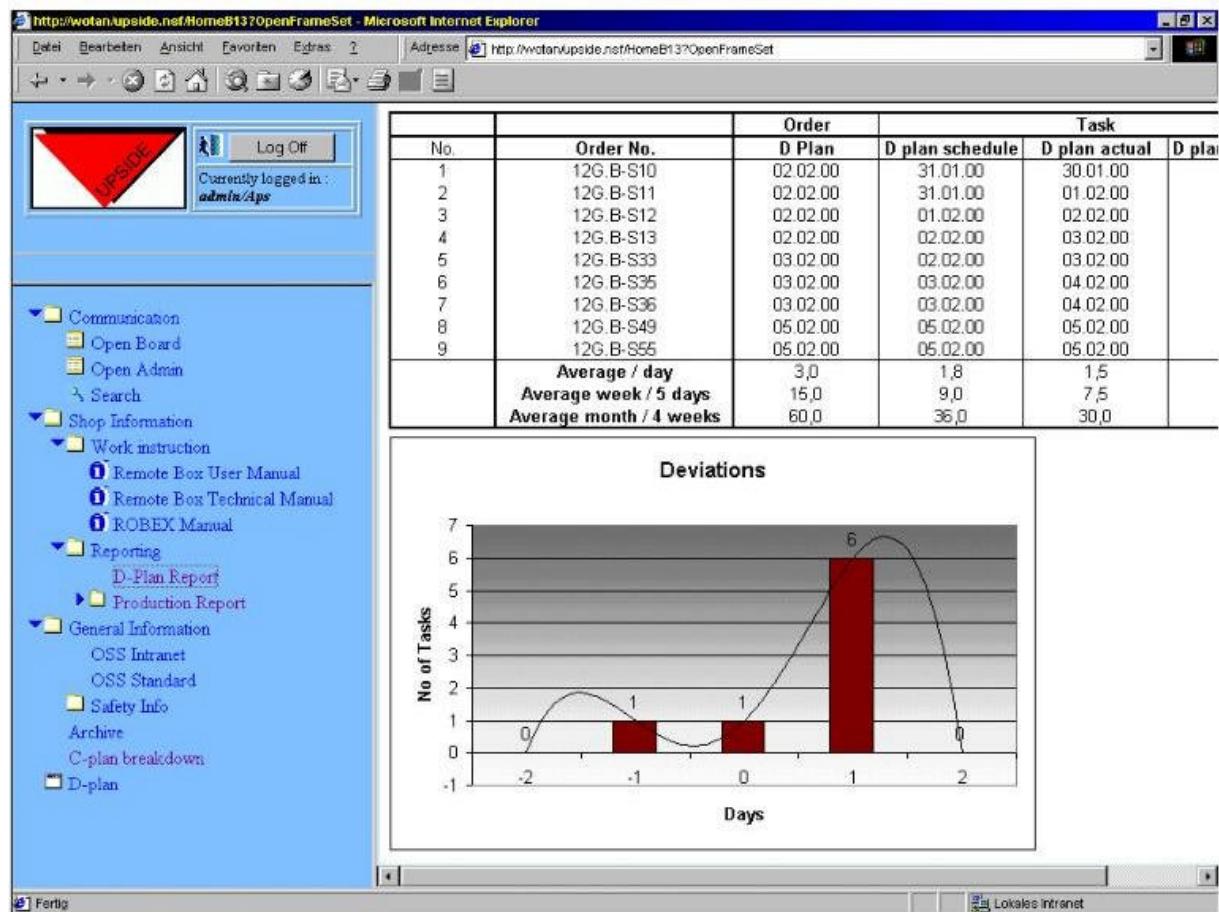


Abbildung 6-22: Evaluierung der Plankonformität im Unterstützungssystem

6.3.5 Kommunikation

Das verteilte Unterstützungssystem soll zusätzlich zur prozessbezogenen Information des Anwenders auch seine Partizipation an Planungs- und Optimierungsvorgängen in der Werkstatt fördern. Ein wesentlicher Aspekt dieser Partizipation ist die Einrichtung von angemessenen Kommunikationsmitteln, die dem Werker Möglichkeiten bieten, z.B. Verbesserungsvorschläge an einzelne Benutzer oder Benutzergruppen zu senden.

Die hier eingesetzte Servertechnik gestattet die Zuordnung einer oder mehrerer Rollen zu jedem Eintrag der Benutzerdatenbank. Ein Benutzer kann beispielsweise sowohl die Rolle „Rohrschmied“ als auch die Rolle „Vorarbeiter“ haben.

Das System ermöglicht prinzipiell, Mitteilungen an

- ➔ einen oder mehrere Anwender,
- ➔ eine oder mehrere durch Rollen definierte Gruppen oder an
- ➔ alle Anwender

zu versenden. Zusätzlich zur oben beschriebenen Möglichkeit wurden Diskussionsbretter integriert, die Anwendergruppen oder allen Anwendern zur Verfügung gestellt werden können.

Die Diskussionsbretter ordnen die Mitteilungen nach Überschrift oder Eingangsdatum. Nach ersten Tests des Systems in der Werkstattumgebung mit einem nicht themenspezifischen Diskussionsbrett regten die Werker aus Übersichtsgründen die Einrichtung von speziellen Diskussionsbrettern für die folgenden Themen an:

- ➔ Allgemeines
- ➔ Material
- ➔ Montage und Wartung von Maschinen
- ➔ Transport
- ➔ Administration

Zur weiteren Verbesserung der Übersicht wurde eine Funktion zur Stichwortsuche implementiert. Die Suchfunktion kann auf bestimmte Bretter beschränkt eingesetzt werden, nach einem bestimmten Autor oder einem bestimmten Thema suchen. Die Ausgabe der Suchergebnisse erfolgt sortiert nach Trefferrelevanz.

6.3.6 Informationssystem

Neben den aktiven Komponenten wurden verschiedene Informationsbausteine in das Unterstützungssystem integriert. Sie dienen der Hilfestellung des Werkers bei schwierigen Aufgaben, der Dokumentation von Standards sowie der Bereitstellung von Informationen über die Ausrüstung der Werkstätten.

Im wesentlichen sind die folgenden Informationen verfügbar:

- ➔ Bedienungsanleitungen von Maschinen und Software der Werkstätten
- ➔ Wartungshandbücher und -pläne
- ➔ Spezifikationen und technische Daten von Maschinen

- ➔ Prozessspezifische Dokumentation wie z.B. Standardisierung des Schweißprozesses
- ➔ Aufgabenbezogene Information

Besonders für unerfahrene Werker und Auszubildende ist die Integration von Informationen dieser Art wichtig und ergänzt so die vorhandenen Aus- und Weiterbildungsprogramme des Unternehmens.

Die technische Integration dieser Informationen wurde für statische bzw. im Verwaltungsbereich der jeweiligen Werkstatt liegenden Daten wie z.B. Maschinenhandbücher oder Schweißaufträge (Abbildung 6-23) durch den Import in den Server DIS.MODUS vollzogen.

Material Requirements		Weld Metal	HAZ
Tensile Test, Strength	N/mm ²	40-66	
Tensile Test, Elongation	%	22 in 50mm	
Additional Requirements, Impact	J	none	none
Additional Requirements, Hardness	HV10	none	none
Additional Requirements	COD	none	none

Groove Preparation: none Cleaning: Initial Aut. grinding Interpass: none

Groove, Type and Definition		Bead Sequence by Numbering	Tacking, Clamping, Backing
		Bead Sequence by Numbering 1.1st, min. 4.5 mm	Tacking, Clamping, Backing 4 tracks are made automat. Hydraulic clamping

Fillermetal	Brand Name	Supplier	Classification
A	ok 12.51	ESAB,Kopenhagen	DIN 8559: SG2 M2 5255
B			

Abbildung 6-23: Spezifikation eines Schweißauftrags

Andere Daten, die entweder dynamisch erzeugt oder von außerhalb der Werkstatt liegenden Stellen administriert werden, wurden als externe Verknüpfungen in das Unterstützungssystem aufgenommen und werden somit für den Anwender transparent im Web-Browser dargestellt.

Beispiel für eine externe Verknüpfung ist das Intranet der Werft, in dem in regelmäßigen Abständen allgemeine Informationen veröffentlicht werden.

Die Administration des internen Informationssystems ist webbasiert möglich, sodass autorisierte Anwender von einem beliebigen PC aus Beiträge einfügen können.

7 Einsatz und Validierung des Systems

7.1 Einsatz in den Werkstätten

Die Validierung des Gesamtkonzeptes erfolgte auf Basis des Einsatzes in der Zielumgebung. Die Schwerpunkte der Betrachtungen liegen in den Bereichen

- ➔ Akzeptanz des Systems,
- ➔ Barrieren bei seiner Einführung und
- ➔ Wirtschaftlichkeit.

Das verteilte Unterstützungssystem wurde mit dem Ziel der Validierung des Gesamtkonzeptes in der Produktion zweier Werkstätten der Werft eingesetzt. Die Basis des Systems ist in beiden Werkstätten identisch, die Implementierungen unterscheiden sich lediglich in der Zusammensetzung der eingesetzten optionalen Komponenten.

7.1.1 Realisierungsbeispiel A: Rohrwerkstatt

Das verteilte Unterstützungssystem wurde in mehreren Werkzellen der Rohrwerkstatt einer Schiffswerft eingeführt. Die Einsatzorte sind im einzelnen:

- ➔ Vorarbeiterbüro
- ➔ Kleine NC-Biegemaschine
- ➔ Große NC-Biegemaschine
- ➔ Manuelle Rohrmontage
- ➔ Halbautomatische Rohrmontage

In Abhängigkeit vom Einsatzort lassen sich unterschiedliche Schwerpunkte bei der Nutzung des Systems feststellen. Während der Vorarbeiter das System hauptsächlich für administrative Aufgaben, zur Kommunikation und Eingabe von Informationen nutzt, wird bei der Rohrmontage am häufigsten das Modul zur 3D-Visualisierung der Rohre verwendet (Abbildung 7-1).



Abbildung 7-1: Arbeitsplatz mit Unterstützungssystem DIS.MODUS

An den Biegemaschinen liegt der Nutzungsschwerpunkt im Bereich der technischen Informationen. Speziell die Dokumentationen zu den Maschinen sowie Parametertabellen für diesen Fertigungsprozess waren von großem Interesse.

7.1.2 Realisierungsbeispiel B: Montagewerkstatt für Strukturelemente

Die zweite Implementierung des Unterstützungssystems fand in der Montagewerkstatt für Strukturelemente statt. Aufgrund der Organisation der Werkstatt und des besonderen Layouts der Produktion wurden hier zunächst nur zwei Arbeitsplätze eingerichtet.

Der erste Arbeitsplatz befindet sich im Vorarbeiterbüro und wird in erster Linie für Planungsaufgaben und – ähnlich wie in der Rohrwerkstatt – zur Administration und

Kommunikation verwendet. Der zweite Arbeitsplatz befindet sich an der Roboterschweißstation der Werkstatt. Auch hier bildet das integrierte Planungssystem den Nutzungsschwerpunkt. Weitere häufig verwendete Funktionen sind die Statusabfrage des Roboters, die Evaluierung der Plankonformität sowie das Kommunikationssystem.

7.2 Akzeptanz des Systems

Die Einführung von informationstechnischen Systemen in teilweise von langjährigen Traditionen geprägte Fertigungsumgebungen stößt an ihre Grenzen, wenn die Akzeptanz dieser Systeme nicht gewährleistet werden kann. Akzeptanz meint hier nicht nur die Akzeptanz des Endanwenders, sondern auch die Akzeptanz durch Arbeiterorganisationen und durch das Unternehmensmanagement.

7.2.1 Akzeptanz durch die Unternehmensleitung

Vor der Entscheidung, ein Informationssystem in eine bestehende Fertigungsumgebung zu integrieren, steht die Akzeptanz solcher Systeme durch die Unternehmensleitung. In der Regel ist die Akzeptanz dann hoch, wenn sich die zusätzlichen Investitionen durch den Einsatz der Informationstechnik relativ kurzfristig durch eine deutliche Steigerung der Produktionseffizienz amortisieren.

Neben dem rein wirtschaftlichen Aspekt gibt es jedoch weitere, nur sehr schwer vorhersehbare Faktoren, die Einfluss auf eine solche Entscheidung haben sollten. Der Mensch, die wichtigste Komponente der Fertigung, wird oft bei der Entscheidungsfindung vernachlässigt. Sein Potenzial kann durch den Einsatz von Informationstechnik freigesetzt, Synergien können hervorgerufen und neues Expertenwissen kann gefördert werden. Motivation, Zufriedenheit und letztlich auch die Attraktivität des Arbeitsplatzes für Neubewerber sind schwer messbare Einflussgrößen, die in diesem Umfeld beachtet werden müssen.

In dem in dieser Arbeit vorgestellten Beispiel hat sich die Unternehmensleitung aus eben diesen Gründen für den Einsatz der Informationstechnik auf Werkstattebene entschieden. Sie fördert aktiv die unternehmensweite Akzeptanz solcher Systeme unter anderem durch regelmäßige Artikel in der Werkszeitung.

7.2.2 Benutzerakzeptanz

Die Akzeptanz durch den Endanwender kann sich als große Barriere bei der Einführung eines Unterstützungssystems herausstellen. Wichtig ist hier, dass das Systemkonzept sich am Anwender orientieren muss. Dem Systemdesign muss eine gründliche Analyse der Anforderungen des Werkers und seiner Aufgaben

vorhergehen. Der Anwender muss sich und seine Aufgabe im System unterstützt sehen, er muss direkte, ihn persönlich betreffende Vorteile in der Nutzung des Systems sehen können.

In beiden Werkstätten, in denen das Unterstützungssystem eingeführt wurde, kam es zu ähnlichen Reaktionen der Werker. Es stellte sich heraus, dass zwischen der Erfahrung des Werkers in seiner jetzigen Tätigkeit und seiner Akzeptanz des Systems eine Korrelation existiert.

Werker, die seit Jahren in derselben Werkstatt, unter Umständen in derselben Werkzelle tätig sind, nutzen das System wenig. Sie vertrauen ihren Erfahrungen und scheuen die zusätzliche Arbeit, die in der Beschäftigung mit dem Unterstützungssystem gesehen wird. Ein anderes Ergebnis stellt man bei den weniger erfahrenen Nutzern fest. Sie sehen das System als Vereinfachung komplexer Prozesse und nutzen seine Funktionen häufiger (Abbildung 7-2).

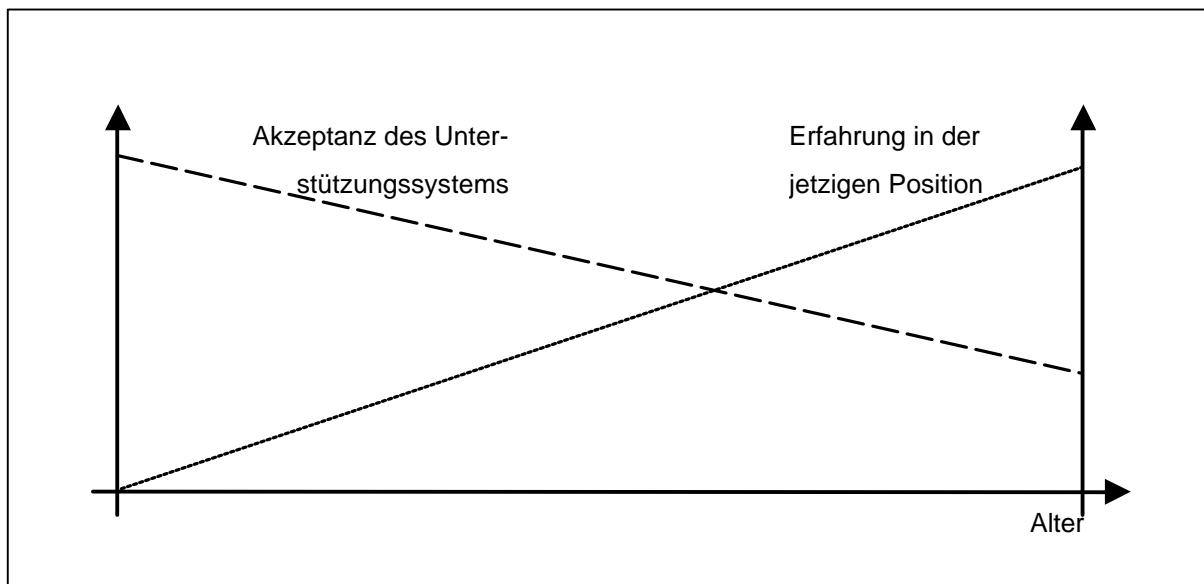


Abbildung 7-2: Akzeptanz des Unterstützungssystems und Erfahrung in Abhängigkeit vom Alter

Da die Arbeitserfahrung der Werker wiederum mit ihrem Lebensalter korreliert, kommen weitere, generationsspezifische Einflussgrößen hinzu. Jüngere Werker sind im allgemeinen stärker an der Arbeit mit Rechnern interessiert als ältere. Oft bringen sie bereits Rechnerkenntnisse aus dem Privatbereich oder aus der Ausbildung mit und wollen diese Kenntnisse einsetzen. Für sie ist die Integration der Informationstechnik eine wichtige Modernisierung ihres Arbeitsplatzes, die sie motiviert und in ihrer Identifikation mit dem Unternehmen stärkt.

Zusammenfassend lässt sich feststellen, dass das Unterstützungssystem zwar schon jetzt von den Anwendern angenommen wird, es sein Potenzial jedoch erst in der Zukunft vollständig umsetzen kann, da es heute zunächst überwiegend von jüngeren

Werkern genutzt wird. Eine weitere Investition in die Zukunft kann in der Steigerung der Attraktivität des Arbeitsplatzes gesehen werden. Ein attraktiver Arbeitsplatz erhöht die Chancen, motivierte und qualifizierte neue Arbeitskräfte zu gewinnen.

7.2.3 Gewerkschaften und Arbeiterorganisationen

Weiterer Gesichtspunkt bei der Validierung des Systems ist die Akzeptanz durch die Gewerkschaften bzw. Arbeiterorganisationen. Bei der Integration des Systems zeigte sich eine skeptische Einstellung der Werker gegenüber dem Planungs- und Evaluierungssystem. Sie sahen in ihm nicht nur eine Möglichkeit, die Arbeit besser zu planen, sondern auch eine neue Kontrollmöglichkeit ihrer persönlichen Effizienz durch den Arbeitgeber. Sie sahen die ihnen durch gewerkschaftliche Absprachen zugesicherten Arbeitszeiten pro Auftrag gefährdet. Es zeigte sich, dass durch eine umfassende Erläuterung des Systems mit der Zusicherung, Daten aus der Evaluierung weder personen- noch teambezogen zu speichern und ausschließlich für die interne Optimierung der Arbeitsplanung zu verwenden, das Vertrauen der Werker teilweise zurückgewonnen werden konnte.

Sehr positiv wurde von den Arbeiterorganisationen die Möglichkeit aufgenommen, öffentliche und geschlossene Diskussionsforen im Kommunikationsmodul des Unterstützungssystems einzurichten. Zusätzlich wünschten sich die Arbeitervertreter eine Sektion im Unterstützungssystem, in der sie Informationen und Termine veröffentlichen können.

7.3 Wirtschaftliche Aspekte

Ein wichtiger Faktor der Akzeptanz eines prozessunterstützenden Systems ist die Einschätzung des Einflusses auf die Wirtschaftlichkeit der Produktion. Kurz- und mittelfristig müssen die Investitionskosten für die Installation und Wartung des Systems in Betracht gezogen werden. In einer ersten Abschätzung geht das Management von einer Reduktion der Produktionskosten um ca. 10% aus. Darin enthalten ist – neben der nur schwer abzuschätzenden Effizienzsteigerung durch Flexibilisierung und die erhöhte Motivation der Werker – auch zu einem großen Teil die Verminderung der Fehlerrate.

Kann das Unterstützungssystem in 20% der Werkstätten eingesetzt werden, ergibt sich daraus eine auf das Unternehmen bezogene Produktivitätssteigerung von 2%. Setzt man die Installationskosten mit 500.000 € (5 Server und 45 Arbeitsplätze) für die gesamte Werft an, so kann die Kapitalrückflussdauer für eine Werft mit 25 Mio. € Vorsteuerjahresgewinn auf ein Jahr geschätzt werden.

Es kann also in einer ersten Schätzung davon ausgegangen werden, dass sowohl die Investitionskosten als auch die hier noch nicht betrachteten Wartungskosten mit großer Sicherheit durch die resultierenden Produktivitätsgewinne abgedeckt werden.

Diese Abschätzung betrachtet noch nicht die längerfristig zu erwartenden Produktivitätsgewinne durch die

- Steigerung der Werkerqualifikation durch Austausch von Wissen,
- Steigerung der Attraktivität des Arbeitsplatzes für qualifizierte Bewerber und
- kontinuierliche Intensivierung der Systemnutzung durch Generationswechsel.

Zusammenfassend lässt sich feststellen, dass sowohl kurz- als auch langfristig die Einführung eines werkerorientierten Unterstützungssystems mit großer Sicherheit eine erfolgversprechende Investition zur Steigerung der Produktivität und des menschlichen Potenzials darstellt.

8 Zusammenfassung und Ausblick

Die fortschreitende Globalisierung der Märkte und der daraus resultierende starke Konkurrenzdruck stellt Unternehmen mit hohem Lohnkostenanteil vor neue Herausforderungen. Insbesondere der europäische Schiffbau sieht sich einem Preiskampf mit asiatischen Werften, die zu wesentlich geringeren Lohnkosten produzieren, ausgesetzt. Gleichzeitig verlangt der Markt Produktinnovationen, die zusätzliche Anforderungen an die Fertigung stellen.

Die in der industriellen Fertigung häufig anzutreffende Automatisierung von manuellen Tätigkeiten bietet aufgrund der im Schiffbau üblichen geringen Losgrößen nur noch begrenzte Möglichkeiten der Kosteneinsparung. Man hat erkannt, dass eine Reorganisation der Arbeitsabläufe, eine Auflösung starrer hierarchischer Strukturen und die Förderung von Teamarbeit ein in Relation zu den Investitionskosten höheres Optimierungs- und Flexibilisierungspotenzial bietet.

Im Rahmen dieser Arbeit wurde ein Konzept zur Nutzung dieses Optimierungspotenzials entwickelt und umgesetzt. Der Ansatz basiert auf der benutzerorientierten Förderung von Kommunikation und Verbreitung von Information auf Werkstattebene und soll den strukturellen Wandel zu einer flexiblen, teamorientierten Arbeitsweise ermöglichen. Das System unterstützt dazu gezielt die Eigenverantwortlichkeit der Worker und stellt entsprechende Planungsfunktionen sowie die für eine Entscheidungsfindung notwendigen Informationen zur Verfügung.

In der Analyse zweier Werkstätten der Odense Stahlschiffswerft wurden Defizite der werkstattinternen Kommunikation und Fertigungsorganisation unter Beachtung der sozialen Randbedingungen untersucht und aufgezeigt. Die bislang in den Werkstätten eingesetzten prozessunterstützenden Systeme sind für die Nutzung an einzelnen Maschinen konzipiert und verfolgen keinen kommunikationsfördernden und informationsintegrierenden Ansatz.

Anhand des Modells des holonischen Fertigungssystems wurden die Grundlagen der Kommunikation in industriellen Werkstätten aufgezeigt. Mithilfe der Anwendung dieses Modells auf die Fertigung im Schiffbau wurden notwendige

Kommunikationskanäle definiert. Diese Definition floss anschließend in die Anforderungen an das Unterstützungssystem ein.

Der Einsatz eines rechnerbasierten Prozessunterstützungssystems stellt besondere Anforderungen in Bezug auf inhaltliche, technische, ergonomische und ökonomische Kriterien. Die Inhalte des Unterstützungssystems sollen die Teamarbeit fördern, den Werker über aktuelle Prozesszustände informieren sowie eine allgemeine Informationsbibliothek enthalten. Aus technischer Sicht muss das System so gestaltet sein, dass es mithilfe einer modularen Architektur unter Verwendung standardisierter Schnittstellen in existierende Infrastrukturen integriert werden kann. Die Anwendung softwareergonomischer Grundsätze soll die Verwendbarkeit auch durch weniger erfahrene Anwender sicherstellen. Durch die Verwendung von Standardplattformen und einfach zu wartenden Systemen können die Investitions- und Betriebskosten für das Unterstützungssystem begrenzt werden.

Die Systemplattform basiert auf einer Client-Server-Architektur mit ausgelagerten echtzeitfähigen Modulen zur Akquisition von Prozessdaten. Durch die Verwendung von Intranettechnologie und Kommunikationsstandards wie ODBC und CORBA wurde eine hohe Integrationsfähigkeit des Systems erreicht. Die Systemplattform ermöglicht somit die Interaktion des Unterstützungssystems mit externen Datenquellen auch in heterogenen Umgebungen, wie sie häufig in industriellen Werkstätten anzutreffen sind.

Die entwickelte Softwarearchitektur basiert auf einem modularen Ansatz, der die Flexibilität und Skalierbarkeit des Gesamtsystems sicherstellt. Somit werden Anpassungen des Systems an andere Werkstätten ermöglicht. Durch die Trennung der Systemebenen Akquisition, Integration und Präsentation wird die Sicherheit und Stabilität des Systems erhöht und die Administration vereinfacht.

Das konzipierte System wurde für zwei Werkstätten der Odense Stahlschiffswerft implementiert. Es vereint verschiedene Funktionalitäten, die bisher auf Werkstattebene nicht verfügbar waren oder nur als spezialisierte Einzellösungen existierten.

Das Unterstützungssystem verfügt über ein offenes, benutzerorientiertes Kommunikationssystem, mit dessen Hilfe der Werker themenorientiert Erfahrungen austauschen und Probleme z.B. mit der Konstruktionsabteilung oder anderen Werkern austauschen kann. Durch die Integration von Standardisierungsinformationen, Prozessspezifikationen, Wartungsplänen und anderen Dokumenten verfügt der Werker über eine Informationsbasis, die er zur Problemlösung heranziehen kann. Die Entwicklung einer echtzeitfähigen Schnittstelle zum Fertigungsprozess ermöglicht die Visualisierung, Überwachung und Regelung von Prozessdaten innerhalb der Benutzeroberfläche sowie die echtzeitnahe

Simulation von automatisierten Maschinen. Der Werker kann somit selbständig Prozesse verifizieren und optimieren. Des weiteren wurden Module für die interaktive 3D-Darstellung von Zielwerkstücken in das Unterstützungssystem integriert.

Die Integration eines werkstattgeeigneten Planungssystems ermöglicht die flexible Projektion von Aufträgen auf Ressourcen. Der Werker kann somit kurzfristige Planabweichungen und Änderungen von Randbedingungen in die Planung einfließen lassen. Zusätzlich wurden verschiedene Module zur Prozessevaluierung entwickelt, die das Lokalisieren von Ursachen von Planabweichungen ermöglichen.

Das System wurde abschließend in den zwei Werkstätten installiert und verifiziert. In ersten Untersuchungen stellte sich bereits heraus, dass vor allem weniger erfahrene und jüngere Werker das System nutzen. Sie haben somit die Möglichkeit, komplexe Aufgaben selbständig zu lösen und sich Expertenwissen anzueignen. Zur vollständigen Einschätzung der resultierenden Effizienzsteigerung in diesen Werkstätten müssen die Ergebnisse laufender Langzeitstudien abgewartet und analysiert werden.

Der Einsatz dieser Technologie stärkt bereits heute die Motivation vor allem jüngerer Arbeitnehmer und erhöht die Attraktivität des Arbeitsumfelds der Werkstatt. Es können somit einfacher junge Kräfte für den Einsatz in der Werkstatt gewonnen werden. Gleichzeitig erhöht sich der Austausch von Wissen und Erfahrung, wodurch die Werkerqualifikation innerhalb des Unternehmens gestärkt wird.

Die offene Architektur des Systems ermöglicht zukünftige Applikationen in anderen, ähnlich strukturierten industriellen Bereichen. Eine weitere Entwicklungsstufe dieser Technologie wäre beispielsweise der Einsatz im Waggonbau oder die Fertigung komplexer Maschinen und Anlagen. Durch die weitere, werkübergreifende Vernetzung der Serverplattform bei Unternehmen mit räumlich getrennten Produktionsstätten könnten in Zukunft zusätzliche Synergieeffekte geschaffen werden.

Mithilfe multimedialer Techniken wie eingebettete Video- oder Audiosequenzen ließe sich der Zugang zu komplexen Informationen weiter verbessern.

Weiteres Optimierungspotenzial ließe sich in Zukunft durch eine Integration der Qualitätssicherung eröffnen. Hier könnten Möglichkeiten zur Informationserfassung und Rückmeldung in die Werkstatt geschaffen werden.

9 Literatur

[ATH98] Atherton, R.:

An Overview on Java Process Control. Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998

[BAC96] Bach, O.:

Abbau von Hierarchien: Mehr Effizienz durch schlankere Strukturen. Sauer-Verlag, Heidelberg, 1996

[BBH97] Brauer, V.; Bruns, F. W.; Hornecker, E.; Robben, B.; Rosch, H.; Rügge, I.; Schäfer, K. (Hrsg.):

Vom Bildschirm zum Handrad. Computer(be)nutzung nach der Desktop-Metapher. Workshop, 6.-7. Oktober 1997, artec-papers Nr.54, Forschungszentrum Arbeit und Technik, Universität Bremen

[BEN99] Benz, S.:

Bahnplanung redundanter Roboterkinematiken mit verteilten Simulationswerkzeugen. Dissertation, RWTH Aachen, 1999

[BEU98] Bergonzi, S.; Uccelli, M. G.:

Why Should the Process Control Industry “Embrace and Expand” Web Technologies. In: Changing the Ways we work, Conference on Integration in Manufacturing, Göteborg, Schweden, 1998

[BDH96] Barber, C. B.; Dobkin, D. P.; Huhdanpaa, H.:

The Quickhull Algorithm for Convex Hulls. ACM Transactions on Mathematical Software, Vol. 22, No. 4, Dezember 1996

- [BON98] Bongaerts, L.:
Integration of Scheduling and Control in Holonic Manufacturing Systems.
Dissertation, Katholische Universität Leuven, Belgien, 1998
- [BOO94] Booch, G.:
Object-oriented Analysis and Design with Applications. *Benjamin Cummings, California, USA, 1994*
- [BRA97] Bradley, M.:
IOP: OMG's Inter-ORB Protocol: A Brief Description. Object Management Group, Needham, USA, 1997
- [BSM99] Bronstein, I. N.; Semendjajew; K. A., Musiol, G.:
Taschenbuch der Mathematik. Verlag Harry Deutsch, 1999
- [BSW93] Beime, J.; Schnidler; R., Wandke; R.:
Wie Experten der Software-Ergonomie den Teil 10 (Dialogue Principles) der ISO 9241 bewerten. In: Rödiger, K.-H. (Hrsg.), Software-Ergonomie – von der Benutzeroberfläche zur Arbeitsgestaltung; Berichte des German Chapter of the ACM; Teubner, 1993
- [BUD97] Buurmann, G. M.; Dustdar, S.:
Telecooperation im Industrial Design – eine Fallstudie des Intranet bei Ford. In: Lehner/Dustdar (Eds.): Telekooperation in Unternehmen. Deutscher Universitäts-Verlag, Gabler-Vieweg, 1997
- [BUS98] Bussmann, S.:
An Agent-oriented Architecture for Holonic Manufacturing Control. First International Workshop on Intelligent Manufacturing Systems IMS-EUROPE, Lausanne, Schweiz, 1998
- [CON98] Condry, M.:
Java and the Industry. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998

- [CUR97] Curtis, D.:
Java, RMI and CORBA. White Paper, Object Management Group,
Needham, USA, 1997
- [CZR97] Czap, H.; Reiter, J.:
Offene und verteilte betriebswirtschaftliche Anwendungssysteme. In:
Lehner/Dustdar (Hrsg.): Telekooperation in Unternehmen. Deutscher
Universitäts-Verlag, Gabler-Vieweg, 1997
- [DEH55] Denavit J.; Hartenberg R. S.:
A Kinematic Notation for Lowpair Mechanisms based on Matrices. In:
ASME Journal of applied Mechanics, Juli 1955
- [DIN66] DIN 66234:
*Teil 8; Bildschirmarbeitsplätze – Grundsätze ergonomischer
Dialoggestaltung,* Beuth-Verlag, Berlin, 1988
- [DIN44] DIN 44300:
Informationsverarbeitung – Begriffe, Beuth-Verlag, Berlin, 1988
- [DIN19] DIN 19245:
PROFIBUS, Process Field Bus, Beuth-Verlag, Berlin, 1991
- [DRE96] Drews, P.; Willms, K.:
*Multisensorsystem zur Erfassung der Prozessumwelt beim
Laserschweißen.* Bänder, Bleche, Rohre S. 36-46, 1996
- [DRE97] Drews, P.; Fromm, P.:
*Multilayer Robot Communication based on a server client gateway
concept.* In: Proceedings of the 1997 IEEE/IES Conference ISIE '97,
Guimaraes, Portugal, 1997
- [DSV97] Drews, P.; Schmid, D.; Volkholz, V.:
Roboter in der Werkstatt. Maschinenbau-Verlag GmbH, 1997

- [EHM97] Ehmke, H.:
Internet and Intranet, In: Lehner/Dustdar (Hrsg.): Telekooperation in Unternehmen. Deutscher Universitäts-Verlag, Gabler-Vieweg, 1997
- [EVA96] Evans, T.:
Building an Intranet. Sams. Net Publishing, 1996
- [FIS96] Fischer, T. M.; Coenenberg, A. G.:
Produktcontrolling im „magischen Dreieck“ von Kosten, Qualität und Zeit. In: Handbuch Betriebshütte; Hrsg.: Eversheim, W.; Schuh, G.; Springer-Verlag; 1996
- [FIS98] Fischer, M.; Stuber, F.:
Work Process Knowledge in Technical Training. In: Changing the Ways we work; Conference on Integration in Manufacturing; Göteborg, Schweden, 1998
- [FLG98] Fabian, M.; Lennartson, B.; Gullander, P.; Andréasson, S.-A.; Adlemo, A.:
Chalmers Architecture and Methodology for Flexible Production – CHAMP. In: Changing the Ways we work; Conference on Integration in Manufacturing; Göteborg, Schweden, 1998
- [FRO95] Fromm, P.; Drews, P.:
Advanced Sensor-Robot Communication. IIW-Jahresversammlung, Doc. XII-1405-95, Stockholm, Schweden, 1995
- [FRR98] Freund, E.; Rokossa, D.; Rossmann, J.:
Process-oriented Approach to an Efficient Off-line Programming of Industrial Robots. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998
- [FRZ91] Frese, M.; Zapf, D.:
Fehler bei der Arbeit mit dem Computer. H. Huber, Göttingen, 1991

- [FRE98] Frese, E.:
Grundlagen der Organisation. Konzept - Prinzipien - Strukturen.
Th. Gabler, Wiesbaden, 1998;
- [FRI97] Frick, O.:
Engineering of Telecoorporation Applications. In: Lehner/Dustdar (Hrsg.):
Telekooperation in Unternehmen. Deutscher Universitäts-Verlag, Gabler-
Vieweg, 1997
- [FUJ00] Website Fujitsu: www.fujitsu.com; zuletzt besucht am 15.09.2000
- [GAL95] Galyga, B.:
Sensorsysteme für die Fertigung von Großkarosserien. Dissertation,
RWTH Aachen, 1995
- [GUB98] Gullander, P.; Bongaerts, L.; Wyns, J.:
*Comparison of Reference Architectures: Chalmers Modified Hierarchical
Architecture vs. K. U. Leuven Holonic Architecture.* In: Changing the Ways
we work; Conference on Integration in Manufacturing; Göteborg,
Schweden, 1998
- [HAO97] Harkey, D.; Orfali, R.:
Client/Server Programming With Java and CORBA. John Wiley & Sons,
1997
- [HAS97] Hallfell, F.; Stammwitz, G.:
Intranets: Offene Informationssysteme im Unternehmen. m&c-
Management & Computer 5/97, pp. 11-18, 1997
- [HEN92] Henrich, D.; Xiaoquing, C.:
*Fast Distance Computation for online Collision Detection with Multi-Arm-
Robots.* IEEE International Conference on Robotics and Automation,
Nizza, Frankreich, 1992

- [HER94] Herczek, M.:
Software - Ergonomie. Grundlagen der Mensch-Computer-Kommunikation. Oldenbourg, München, 1994
- [HMS99] Hienz, H.; Marrenbach, J.; Steffan, R.; Akyol, S.:
Multimodal Human-Computer Communication in Technical Applications.
In Bullinger, H.J., Ziegler, J. (Hrsg.): *Human-Computer Interaction: Ergonomics and User Interfaces, Proceedings of the HCI International '99*, München, pp. 755-759, Lawrence Erlbaum Associates Publishers, London, Großbritannien, 1999
- [HÜS94] Hüsener, T.:
Entwurf komplexer Echtzeitsysteme – State of the Art. In: *Angewandte Informatik*, Band 11, BI-Wissenschaftsverlag, 1994
- [IMA97] Imai, M.:
Gemba Kaizen – Permanente Qualitätsverbesserung, Zeitersparnis und Kostensenkung am Arbeitsplatz. Wirtschaftsverlag Langen Müller / Herbig, 1997
- [ISO93] ISO-IS 11898:
Road vehicles - Interchange of digital information - Controller Area Network (CAN) for high speed communication, 1993
- [JAC98] Jack, H.:
Introduction to Kinematics of Mechanisms. Grand Valley State University (GVSU), 1998
- [JEN97] Jenkins, M. S.:
Abstract Data Types in Java. Computing McGraw-Hill, Juni 1997
- [JOH98] Johansson, A.; Harlin, U.:
Balancing Human Aspects in Dynamic Automated Manufacturing Systems. In: *Changing the Ways we work; Conference on Integration in Manufacturing;* Göteborg, Schweden, 1998

- [JRO94] Rostock, J.:
Aufbau eines hierarchisch organisierten, wissensunterstützten Fertigungsregelsystems. Dissertation, Universität des Saarlandes, 1994
- [KAM98] Kampker, M.:
New Ways of User-Oriented Robot Programming. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998
- [KAM00] Kampker, M.:
Werkzeuge für die verbesserte, nutzerorientierte Werkstattprogrammierung von Schweißrobotern. Dissertation, RWTH Aachen, 2000
- [KIG98] Kirkemo, K.; Getty, R.:
High Performance Teams. In: Changing the Ways we work; Conference on Integration in Manufacturing; Göteborg, Schweden, 1998
- [KOE89] Koestler, A.:
The Ghost in the Machine. Arkana Books, 1989
- [LIC91] Lin, M.; Canny, J. F.:
Efficient Algorithms for incremental Distance Computation. IEEE Conference on Robotics and Automation, 1991
- [LÜT98] Lüth, T.:
Technische Multi-Agenten-Systeme – Verteilte autonome Roboter- und Fertigungssysteme. Carls Hanser Verlag, 1998
- [MAC97] Mackay, R.:
IT and Changing Manufacturing Paradigms. Proceedings of the Workshop “Human Aspects in Shopfloor Environments”, Europäisches Centrum für Mechatronik, Aachen, April 1997

- [MAT97] Matzner, D.:
Modulare Steuerungskomponenten für Industrieroboter mit kinematischer Redundanz. Dissertation, RWTH Aachen, 1997
- [MEC95] Mechler, B.:
Intelligente Informationssysteme. Addison-Wesley, 1995
- [MEG93] Mertens, P.; Giese, J.:
Integrierte Informationsverarbeitung 2 – Planungs- und Kontrollsysteme in der Industrie. Gabler, 1993
- [MML95] Merz, M.; Müller-Jones, K.; Lamersdorf, W.:
Middleware Support for Open Distributed Applications. In: Tagungsband First International Workshop on High Speed Networks and Open Distributed Platforms, St. Petersburg, Juni 1995
- [MON97] Ott, M.; Nastansky, L.:
Verteilung und Kooperation bei der Modellierung von Workflow Management. In :Telekooperation in Unternehmen, Hrsg.: Lehner, F.; Dustdar, S., Deutscher Universitäts-Verlag, Wiesbaden, 1997
- [NAH95] Nastansky, L.; Hilpert, W.:
Das GroupFlow System für Workflow-Management, Balance zwischen Struktur und Flexibilität. Business Computing Nr.7, Vogel-Verlag, Würzburg, 1995
- [NEI96] Neider, J.:
OpenGL Programming Guide. Addison Wesley, 1996
- [NOE98] Nordqvist, A. R.; Ericsson, P. T.:
Design of Distributed Adaptive User Interfaces Using Digital Plant Technology. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998
- [OBS95] N. N.:
Proceedings of the Open Bus Systems, Zürich, 11.-13.10.1995

- [OHW98] Ort, M.; Hemmerling, B.; Wiedemann, T.:
Intranet-based Workflow Management System. Universität Potsdam - Workshop: Soziotechnischer Zugang bei der Planung, Einführung und Anwendung von Workflowmanagement-Systemen, Preprint 002, Potsdam, 1998
- [OSA97] Lutz, P.; Sperling, W.:
OSACA – The Vendor neutral Control Architecture. In: Facilitating Development of Information and Communications Technologies for Competitive Manufacturing, Proceedings of the European Conference on Integration in Manufacturing liM'97, Dresden, Selbstverlag der TU Dresden, 1997
- [OSI98] N.N.:
Introduction into the OSACA Architecture. OSACA Consortium, 1998
- [OSP98] N.N.:
Programming OSACA Applications. OSACA Consortium, 1998
- [OUS94] Ousterhout, J.:
Tcl and the Tk Toolkit. Addison-Wesley, 1994
- [PAN00] Website Panasonic: www.panasonic.com; zuletzt besucht am 15.09.2000
- [PAO99] Ort, M.; Pape, U.:
Flexible Process Management. In: Bernd Scholz-Reiter, Process Modelling, Springer-Verlag, 1999
- [PDC98] N. N.:
CAPS – Computer-Aided Process Scheduling, Introduction. Prolog Development Center, Brøndby, Dänemark, 1998
- [PDT99] Productivity Development Team:
Cellular Manufacturing: One-Piece Flow for Workteams. Productivity Press Inc., 1999

- [PRO00] Website Protech: www.protech.com; zuletzt besucht am 15.09.2000
- [QUA95] Quartier, F.:
Ein Beitrag zur Nutzerorientierten Bedienung und rechnergestützten Werkstattprogrammierung von Schweißrobotern. Dissertation, RWTH Aachen, 1995
- [REI95] N.N.:
Technische Beschreibung der Robotersteuerung ROBOTstar IV. Reis Robotics, 1995
- [RSW98] Richardson, T.; Stafford -Fraser, Q.; Wood, K. R.; Hopper, A.:
Virtual Network Computing. Olivetti & Oracle Research Laboratory, Cambridge, Großbritannien, in: IEEE Internet Computing, Vol. 2, No. 1, Januar/Februar 1998
- [RUM91] Rumbaugh, J.:
Object-oriented Modelling and Design. Prentice-Hall, New Jersey, USA, 1991
- [RYA98] Ryan, M. R.:
The QUAKIT Service Package: Organisational Change Management, Software- and Paper-Based Tools, Case Study Library, Training Packages and Consultancy Services. In: Changing the Ways we work, Conference on Integration in Manufacturing, Göteborg, Schweden, 1998
- [SCH96] Schill, A.:
Distributed Platforms. In: Encyclopedia of Microcomputers, Marcel Dekker Publishing, 1996
- [SCH97] Schulten, M.:
PC-basiertes, interaktives Programmiersystem für Industrieroboter. Diplomarbeit RWTH Aachen, 1997

- [SCH98] Schulten, M.; Benz, S.:
User-Oriented Production Support. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998
- [SCH00] Schulten, M; Bjerregaard, J.; Rast, H.:
User-oriented Production Support in Distributed Shop-Floor Environment. Public Final Report; ESPRIT-Projekt 22626; Brüssel, 2000
- [SCR98] Schreck, G.:
Advanced Man-Machine-Interfaces for Robot System Applications. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998
- [SCW95] Schulze-Wehninck, F.-J.:
Sensorsysteme zur Führung von Industrierobotern – Aufbau, Anwendung und Simulation. Dissertation, RWTH Aachen, 1995
- [SEP98] Sepet, N.:
Controlling of Scheduled Production Processes. In: Proceedings of the 31st CIRP International Seminar on Manufacturing Systems "Networked Manufacturing: Integrated Design, Prototyping and Rapid Fabrication", pp. 189-194, Berkeley, Kalifornien, USA, 1998
- [SKL97] Schmidt K.-H.; Kleinbeck, U.:
Relationships between group-based performance measures, feedback, and organizational context factors. European Journal of Work and Organizational Psychology, 6 (3), pp. 303-319, 1997
- [SOL93] Solvie, M.:
FIP - die französische Lösung. pa - Produktionsautomatisierung 1/93, 1993
- [SPI87] Spinas, P.:
Arbeitspsychologische Aspekte der Benutzerfreundlichkeit von Bildschirmsystemen. Dissertation, ETH Zürich, 1987

- [STE00] Steflik, D.; Sridharan, P.; steflik, R.:
Advanced Java Networking. Prentice Hall, 2000
- [STV94] Stadelmeyer, V.:
Entscheidungsunterstützung zur technischen Planung im Fertigungsbereich. Dissertation, Universität des Saarlandes, 1994
- [TEC99] N. N.:
RobCAD Users Guide, Tecnomatix. Dietzenbach, 1999
- [TLM98] Teunis, G.; Leitão, P.; Madden, M.:
A New Architecture for Flexible Shop Control Systems. In: Changing the Ways we work; Conference on Integration in Manufacturing, Göteborg, Schweden, 1998
- [TSM95] Teufel, S.; Sauter, C.; Mühlherr, T.; Bauknecht, K.:
Computerunterstützung für die Gruppenarbeit. Addison-Wesley, Bonn 1995
- [TWW95] Tharumarajah, A.; Wells, A. J.; Wirth, A.:
Scheduling in Distributed Production Systems. ICME conference, Melbourne, Australia, 1995
- [UNO99] United Nations Economic Commission for Europe / Statistical Division; World Robotics 1999:
Press Release. Genf, 1999
- [VER98] Verjans, S.:
IT-Supported Organisational Changes in a Production Shop Floor: A Case Study. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998
- [VET98] Vetter, M.:
Aufbau betrieblicher Informationssysteme mittels pseudo-objektorientierter, konzeptioneller Datenmodellierung. B. G. Teubner Stuttgart, 1998

- [VIN97] Vinoski, S.:
CORBA - Integrating diverse applications within distributed heterogeneous environments. In: IEEE Communications Magazine, Vol. 14, No. 2, 1997
- [VIT95] N. N.:
VMEbus Products Directory,
VMEbus International Trade Association (VITA), Phoenix, Arizona, 1995
- [VML98] Verjans, S.; Mogensen, L. K.; Lynngard, H.-J. B.:
Interactive Collaborative Production Scheduling. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, 1998
- [VSM00] Verband für Schiffbau und Meerestechnik e.V.:
Weltschiffbau 1999. Hamburg, 2000
- [W3H00] W3C – The World Wide Web Consortium:
HTML Specification. Laboratory for Computer Science, Massachusetts Institute of Technology, 2000
- [WAA96] Warnecke, G.; Augustin, H.:
Knowledge Based Information Design of Network Production Segments. Production Engineering, Vol. III/2, pp. 119-122, 1996
- [WET96] Wendel, T.:
Computerunterstützte Teamarbeit, Konzeption und Realisierung eines Teamarbeitssystems. Gabler Edition Wisenschaft, Deutscher Universitäts-Verlag, Wiesbaden, 1996
- [WHS94] Westkämper E.; Höpf M.; Schaeffer C.:
Holonic Manufacturing Systems (HMS). In: Holonic Manufacturing Systems (HMS) Plenary, HMS Consortium, Test Case 5 Proceedings, Lake Tahoe, Kalifornien, USA, 1994

- [WKG97] Watson, R. T.; McKeown P. G.; Garfield, M.:
Topologies for Electronic Cooperation. In: Lehner/Dustdar (Hrsg.):
Telekooperation in Unternehmen. Deutscher Universitäts-Verlag, Gabler-
Vieweg, 1997
- [WLO92] Wloka, D. W.:
*Robotersysteme 1: Technische Grundlagen; Robotersysteme 2:
Graphische Simulation.* Springer-Verlag, 1992
- [WSH98] Warnecke, G.; Stammwitz, G.; Hallfell, F.:
Intranets als Plattform für Groupware-Anwendungen. Industrie-
Management 14, pp. 24-28, 1998
- [YAD96] Yang, Z.; Duddy, K.:
*CORBA: A Platform for Distributed Object Computing (A State-of-the-Art
Report on OMG/CORBA).* Operating Systems Review 30, pp. 4-31, 1996
- [YAS98] Yang, Z.; Sun, C.:
CORBA For Hard Real Time Applications: Some Critical Issues. Operating
Systems Review 32, pp. 64-71, 1998
- [ZIV98] Zirkelbach, C.; Verbarg, K.:
Databases in Automated Process Control. In: Proceedings of the 24th
Annual Conference of the IEEE Industrial Electronics Society, Aachen,
1998

Lebenslauf

Persönliche Daten

Name	Martin Schulten
Geburtsdatum und -ort	12.07.1969 in Krefeld
Staatsangehörigkeit	deutsch
Familienstand	ledig

Schulausbildung

1975 - 1979	Grundschule Oberbruchstraße, Krefeld
1979 - 1988	Maria-Sybilla-Merian-Gymnasium, Krefeld Abschluss: Allgemeine Hochschulreife

Wehrdienst

1988 - 1989	Grundwehrdienst in einer Instandsetzungsausbildungskompanie, Coesfeld
-------------	---

Studium

1989 - 1997	Studium der Elektrotechnik an der RWTH Aachen, Vertiefungsrichtung „Nachrichtentechnik“, Abschluss als Diplom-Ingenieur
-------------	---

Berufstätigkeit

1989 - 1997	Insgesamt 26 Wochen Praktikum im Ingenieurbüro Siempelkamp Pressensysteme, Krefeld, sowie bei der Firma Benedens Sondermaschinenbau, Krefeld
1992 - 1995	Studentische Hilfskraft in der Abt. für Prozeßsteuerung in der Schweißtechnik, RWTH Aachen
Mai 1997 – Dez. 2000	Projektleiter bei der APS GmbH / Europäisches Centrum für Mechatronik, Aachen Leitung: Univ.-Prof. Dr.-Ing. Paul Drews
seit Februar 2001	Senior Technical Consultant bei der Antwerpse & Partner AG, Köln

Köln, im Mai 2003

