



Thema:

Erarbeitung eines Datenbank-Konzeptes für ein Werkzeug zur
Planung und Visualisierung von Fabrikstrukturen

Bachelorarbeit

Fakultät für Informatik
Institut für Technische und Betriebliche Informationssysteme (ITI)
Arbeitsgruppe Datenbanken

Themensteller: Prof. Dr. rer. nat. habil. Gunter Saake
Betreuer: Dipl.-Inform. Andreas Lübcke
Externer Betreuer: Dipl.-Ing. René Petri
Dipl.-Wirtsch.-Ing. Thomas Dengler
vorgelegt von: Konstanze Winter
Matrikelnummer: 179336
E-Mail: Konstanze.Winter@Student.Uni-Magdeburg.de
Abgabetermin: 17. November 2010

Inhaltsverzeichnis

Inhaltsverzeichnis	III
Abkürzungsverzeichnis	V
Abbildungsverzeichnis	VII
Tabellenverzeichnis	IX
1 Einleitung.....	1
1.1 Problemstellung.....	1
1.2 Aufgabenstellung.....	1
1.3 Vorgehensweise.....	2
2 Grundlagen von Datenbanken	3
2.1 Datenbanksysteme.....	3
2.2 Datenbankentwurf	5
2.3 Die konzeptionelle Phase – Das ER-Modell	7
3 Quovadis – Werkzeug zur Unterstützung der Konzeptplanung im Fabrikplanungsprozess.....	9
3.1 Fabrikplanung.....	9
3.1.1 Phasen der Fabrikplanung	10
3.1.2 Konzeptplanung	12
3.2 Funktionsweise des Werkzeugs.....	13
3.3 Analyse der vorhandenen Datenbankanbindung und der Struktur der Datenbank.....	18
4 Datenbank-Konzept	21
4.1 Ableitung notwendiger Anforderungen - Anforderungsanalyse.....	21
4.1.1 Informationsanforderungen.....	21
4.1.2 Verarbeitungsanforderungen.....	34
4.2 Konzeptioneller Entwurf	35
4.3 Logischer Entwurf.....	36
5 Evaluierung.....	41
5.1 Umsetzung der Anforderungen	41
5.2 Normalform	42
6 Zusammenfassung und Ausblick	45
Literaturverzeichnis	XI
Abschließende Erklärung	XIII

Abkürzungsverzeichnis

BLOB	B inary L arge O bject
CLOB	C haracter L arge O bject
DBMS	D aten b ank m anagementsystem
ER	E ntity- R elationship
IFF	Fraunhofer-Institut für Fabrikbetrieb und -automatisierung
Quovadis	Q uickly configurable v irtual i ndustry scenarios
SQL	S tructured Q uery L anguage
THM	T ransporthilfsmittel
VDT	V irtual D evelopment and T raining

Abbildungsverzeichnis

Abbildung 1-1: Vorgehensweise und Aufbau der Bachelorarbeit.....	2
Abbildung 2-1: Relation [SSH08, Seite 86]	4
Abbildung 2-2: Phasen des Datenbankentwurfs [SSH08, Seite 124].....	6
Abbildung 2-3: Entity, Beziehung, Attribut	7
Abbildung 2-4: Beziehungen [SSH08, Seite 72 f.]	8
Abbildung 3-1: Planungsebenen [Sch10, Seite 12]	10
Abbildung 3-2: Fabrikplanungsprozess.....	11
Abbildung 3-3: Konzeptplanung	12
Abbildung 3-4: Strukturierungsprinzipien [Poe08, Seite 9].....	13
Abbildung 3-5: Funktionsweise von Quovadis [Sch10, Seite 9].....	14
Abbildung 3-6: Planungsrelevante Daten.....	14
Abbildung 3-7: Erzeugnisstruktur für ein Fahrrad	15
Abbildung 3-8: ER-Modell des Prototyps	19
Abbildung 4-1: Konzeptioneller Entwurf.....	35

Tabellenverzeichnis

Tabelle 1: Gegenüberstellung der Bezeichnungen	11
Tabelle 2: Datentypen.....	22
Tabelle 3: Entity Werk	22
Tabelle 4: Entity Betrieb.....	23
Tabelle 5: Entity Etage	23
Tabelle 6: Entity Funktionsbereich.....	24
Tabelle 7: Entity Betriebsmittel.....	25
Tabelle 8: Entity Technologie	26
Tabelle 9: Entity Erzeugnis	27
Tabelle 10: Entity Arbeitsplan.....	28
Tabelle 11: Entity Dokument	29
Tabelle 12: Entity Berechnungsergebnisse.....	29
Tabelle 13: Entity Szenario	30

1 Einleitung

Die Planung einer Fabrik ist für den Erfolg eines Unternehmens wichtig und erfolgt auf lange Sicht. In diesem Kapitel werden zunächst die Probleme durch den Wandel des Marktes aufgezeigt und die Ziele sowie die Vorgehensweise dieser Arbeit beschrieben.

1.1 Problemstellung

Unternehmen sind heute wechselnden Einflüssen und Wandlungen unterworfen [Gru06, Seite 12]. Dadurch sind sie gezwungen konsequent kunden- und wertschöpfungsorientiert zu denken [Gru06, Seite 12]. Um sich der steigenden Marktdynamik und dem erhöhten Kostendruck anpassen zu können, muss die Fabrik wirtschaftlich, flexibel, variabel und attraktiv sein [Gru06, Seite 5,10]. Durch die zunehmende Konkurrenz an den Märkten sind die Erwartungen der Kunden gestiegen. Sie fordern Produkte, die preisgünstig, qualitativ hochwertig und ihren individuellen Bedürfnissen zugeschnitten sind [Gru06, Seite 11].

Die Erwartungen der Kunden ändern sich fortlaufend, wodurch sich der Produktlebenszyklus verkürzt. Das führt dazu, dass sich die Produktentwicklungs- und -einführungszeit für das Unternehmen reduzieren [Gru06, Seite 11]. Stetig neu entwickelte Technologien und Werkstoffe im Produktionsprozess erhöhen die Komplexität der Fabriken und Maschinen. Hierdurch werden die Aufgaben der Fabrikplanung zu einem kontinuierlichen Prozess.

1.2 Aufgabenstellung

Um den Fabrikplanungsprozess zu unterstützen, werden computergestützte Systeme verwendet. Am Fraunhofer-Institut für Fabrikbetrieb und –automatisierung (IFF) in Magdeburg wird ein Werkzeug entwickelt, das die Planung und Visualisierung von Fabrikstrukturen sowie die Verwaltung von Planungsdaten unterstützt. Der Prototyp dieses Werkzeugs, mit dem Namen Quovadis (**quickly configurable virtual industry scenarios**), wurde im Rahmen einer Diplomarbeit konzeptionell weiterentwickelt und mit zusätzlichen Planungs- und Visualisierungsfunktionen ausgestattet.

Die Aufgabe dieser Bachelorarbeit besteht darin, die Gestaltung und Struktur der Datenbankbindung, ausgehend vom Prototyp und der konzeptionellen Weiterentwicklung, zu analysieren. Darauf aufbauend wird ein Datenbank-Konzept für die Erweite-

rung entwickelt. Während des Planungsprozesses werden verschiedene Szenarien generiert. Diesbezüglich wird untersucht, welche Daten zur Speicherung benötigt werden und wie die Speicherung erfolgen kann. Neben Szenarien sollen auch Grafiken, Tabellen sowie Zeichnungen in der Datenbank abgelegt werden. Durch diese Analysen werden Anforderungen definiert und ein Konzept für die Datenbank erarbeitet.

1.3 Vorgehensweise

Diese Arbeit gliedert sich in sechs Kapitel. Das erste Kapitel beschreibt die Problemstellung, die Aufgabenstellung sowie die Vorgehensweise der Arbeit, die in Abbildung 1-1 gezeigt wird.

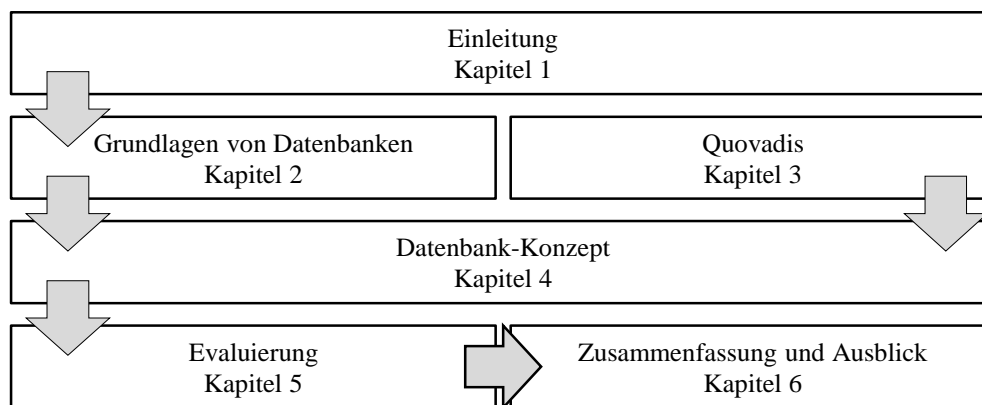


Abbildung 1-1: Vorgehensweise und Aufbau der Bachelorarbeit

Kapitel 2 beinhaltet eine umfassende Beschreibung der Grundlagen von Datenbanksystemen und deren Vorteil gegenüber herkömmlichen Dateisystemen. Das Phasenmodell für den Datenbankentwurf dient als Basis für den Aufbau des vierten Kapitels. Eine Einführung in die Fabrikplanung wird im dritten Kapitel gegeben und der Aufbau sowie die Funktionsweise des Werkzeugs Quovadis erklärt. Weiterer Bestandteil des dritten Kapitels ist die Analyse der vorhandenen Datenbank und deren Struktur.

Im vierten Kapitel werden die Anforderungen an die Datenbank und die Modellierung der Entities und ihrer Beziehungen vorgenommen. Mit der Evaluierung im fünften Kapitel wird die Richtigkeit des Datenbankentwurfs überprüft. Abschließend werden in Kapitel sechs eine Zusammenfassung der Arbeit und ein Ausblick auf weitere Entwicklungsmöglichkeiten gegeben.

2 Grundlagen von Datenbanken

In diesem Kapitel werden die Grundlagen von relationalen Datenbanksystemen und deren Vorteil zur Speicherung von Daten näher erläutert. Es wird außerdem eine Möglichkeit zur Modellierung von Objekten und ihren Beziehungen mit dem Entity-Relationship-Modell (ER-Modell) aufgezeigt sowie das Phasenmodell für den Datenbankentwurf nach [SSH08, Seite 123] vorgestellt, welches für das Datenbank-Konzept in Kapitel 4 verwendet wird.

2.1 Datenbanksysteme

Ein Datenbanksystem besteht aus einer Datenbank und einem Datenbankmanagementsystem (DBMS) und dient zur Speicherung und Verwaltung von Daten. Als Datenbank wird der strukturierte, von einem DBMS verwaltete Datenbestand bezeichnet. Das DBMS enthält die Struktur in der die Daten abgelegt werden und umfasst die gesamten Softwaremodule zur Verwaltung der Datenbank [SSH08, Seite 8 f.].

Der Vorteil von Datenbanksystemen gegenüber Dateisystemen ist die einheitliche Verwaltung aller von den Anwendungen benötigten Daten [Kud07, Seite 23]. So werden Datenredundanzen (mehrfache Speicherung der gleichen Daten) vermieden, indem jedes logische Datenelement, z.B. ein Kundenname, an nur einer Stelle in der Datenbank gespeichert wird [Kud07, Seite 23]. Werden die Daten redundant hinterlegt, muss eine Aktualisierung möglicherweise mehrmals ausgeführt werden [EN02, Seite 35]. Dadurch wird Speicherplatz verschwendet und es entstehen Inkonsistenzen, da eine Aktualisierung in bestimmten, aber nicht in allen relevanten Dateien ausgeführt wird [EN02, Seite 35]. Jedoch ist in manchen Fällen die redundante Speicherung von Daten erforderlich, um z.B. Datenverlust zu vermeiden oder die Effizienz von Anfragen zu verbessern [EN02, Seite 35]. Zudem bietet ein DBMS die Möglichkeit komplexe Beziehungen zwischen den Daten zu definieren sowie zusammenhängende Daten schnell und effizient zu verknüpfen [Kud07, Seite 23]. Werden die Daten zentral gehalten, sind die Zugriffskontrolle und Datensicherheit nicht gewährleistet [SSH08, Seite 3]. Ein DBMS benutzt des Weiteren Techniken zum effizienten Speichern und Auslesen großer Mengen von Daten [Kud07, Seite 24 f.]. Die hohe Flexibilität von Datenbanksystemen ist ein weiterer Vorteil, da die Struktur der Datenbank bei sich ändernden Anforderungen modifiziert werden kann [Kud07, Seite 24 f.].

Es gibt verschiedene Formen von Datenbanksystemen, die sich in der Art und Weise der Datenspeicherung unterscheiden. Es werden z.B. das hierarchische, Netzwerk-, relationale, objektrelationale und objektorientierte Datenbankmodell unterschieden [Kud07, Seite 22; SSH, Seite 511 ff.]. In der Praxis wird überwiegend das relationale Datenmodell eingesetzt [Kud07, Seite 22]. Da das relationale Datenmodell Bestandteil des Konzeptes ist, wird im Folgenden auf Aufbau und Vorteile relationaler Datenbanken eingegangen.

Konzeptuell ist eine relationale Datenbank eine Menge von Tabellen (Relationen), die in Beziehung zueinander stehen und über Schlüssel verknüpft werden. Jede Tabelle speichert Informationen über ein Objekt. Als Datensatz (Tupel) wird eine Zeile der Tabelle bezeichnet. Eine Relation ist die Menge der Datensätze. Jeder Datensatz besitzt Attribute (Eigenschaften). Die Menge der Attribute wird als Relationenschema bezeichnet (Abbildung 2-1).

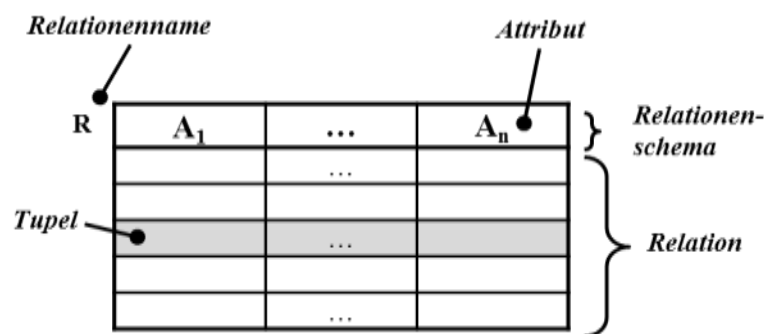


Abbildung 2-1: Relation [SSH08, Seite 86]

Jeder Datensatz ist über ein Attribut bzw. eine Attributkombination identifizierbar. Wenn eine Attributmenge eine Relation eindeutig identifizieren kann, besitzt sie die Schlüsseleigenschaft und wird als Primärschlüssel bezeichnet [Kud07, Seite 77]. In einer Tabelle darf das Attribut bzw. die Attributkombination nur einmal vorkommen. Wird der Primärschlüssel durch eine andere Relation referenziert, wird er Fremdschlüssel genannt. Er dient als Verweis zwischen zwei Relationen und ist relevant, wenn diese miteinander verknüpft werden. Der Fremdschlüssel enthält die gleichen Werte wie der Primärschlüssel in der referenzierten Relation, kann aber öfter vorkommen. Eine Relation kann mehrere Fremdschlüssel besitzen, die verschiedene Relationen referenzieren [Kud07, Seite 77 ff.].

Der Vorteil eines relationalen Datenbanksystems ist, dass seine Struktur geändert (Tabellen hinzufügen, löschen usw.) werden kann, ohne die Anwendung ändern zu müssen. Das ist auch das Ziel der Datenunabhängigkeit. Um diese in Datenbanksystemen zu

unterstützen, wurde die Drei-Ebenen-Schemaarchitektur entwickelt, die ein Datenbank-schema in ein internes, konzeptuelles und externes Schema aufteilt [SSH08, Seite 30-32]. Des Weiteren können viele Sichten oder virtuelle Tabellen erstellt werden, ohne Änderungen an der physischen Datenbank vornehmen zu müssen.

2.2 Datenbankentwurf

Beim Datenbankentwurf ist es das Ziel Informations- und Konsistenzerhalt zu gewährleisten. Informationserhalt sagt aus, dass in der entworfenen Datenbank alle Informationen des ursprünglichen Modells, z.B. des ER-Modells, gespeichert werden können. Im Relationenmodell entspricht dies den Begriffen der Verbund- und Abhängigkeitstreue. Verbundtreue drückt aus, dass eine durch Anwendung der Normalformen zerlegte Relation mittels eines Verbundes wieder in die Originalrelation überführt werden kann [SSH08, Seite 120]. Dabei können jedoch Verluste entstehen. Abhängigkeitstreue ist in diesem Zusammenhang wichtig, da auch die gesamten Abhängigkeiten im Datenbankschema repräsentiert werden sollen [SSH08, Seite 183]. Konsistenzerhalt ist erfüllt, wenn in der Modellierung alle Regeln und Einschränkungen gewährleistet sind, die in den Anforderungen gefordert waren. Im Relationenmodell beschreibt der Begriff Abhängigkeitstreue dieses Merkmal [SSH08, Seite 120].

Das Modell in Abbildung 2-2 stellt ein mögliches Phasenmodell für den Entwurf einer Datenbank dar. Da es kein streng sequentielles Modell ist, sind Korrekturen und Verbesserungen in vorherigen Phasen möglich [SSH08, Seite 123].

Die **Anforderungsanalyse** umfasst eine informale Beschreibung des Problems. Hier werden zunächst alle Informationen und Anforderungen gesammelt. Dann werden die Informationen über Daten (Datenanalyse) von den Informationen über Funktionen (Funktionsanalyse) getrennt. Die Informationen stammen von den späteren Anwendern oder aus den Fachabteilungen und müssen für die Aufgabenstellung gefiltert werden. Das Ergebnis der Anforderungsanalyse ist eine informelle Beschreibung der erforderlichen Prozesse und des Datenbedarfs [SSH08, Seite 122 ff.].

In der zweiten Phase des **konzeptionellen Entwurfs** wird eine erste formale Beschreibung des Fachproblems vorgenommen. Es soll ein zielsystemunabhängiges Datenbankschema erstellt werden. Mit Hilfe des Schemas werden die Daten, die in der Datenbank gespeichert werden sollen, modelliert. Das Ergebnis dieser Phase ist ein semantisches

Datenmodell, wie z.B. das ER-Modell, das in Kapitel 2.3 behandelt wird [SSH08, Seite 122 ff.].

Ein **Verteilungsentwurf** ist nötig, wenn die Datenbankanwendung verteilt auf verschiedenen Systemen realisiert werden soll [SSH08, Seite 127].

In der Phase des **logischen Entwurfs** wird das konzeptionelle Schema in ein relationales Modell transformiert. Das Ergebnis dieser Phase ist das logische Schema, z.B. das Relationenschema [SSH08, Seite 128 f.].

In der **Datendefinition** wird das logische Schema in ein konkretes Schema, unter Verwendung der Datendefinitions- und Datenmanipulationssprache eines implementierten DBMS, umgesetzt [SSH08, Seite 129]. Die Sprache bei relationalen DBMS ist die Structured Query Language (SQL).

Die Angabe konkreter Speicherungsstrukturen erfolgt im **physischen Entwurf**. Es werden außerdem Attribute ausgewählt, die Selektionskriterien in Anfragen darstellen und für die ein zusätzlicher Suchindex angelegt wird. Unter Umständen muss das System an neue Anforderungen und Systemplattformen angepasst werden [SSH08, Seite 130].

In der Phase **Implementierung und Wartung** wird die Datenbankanwendung installiert und an neue Anforderungen angepasst [SSH08, Seite 123]. Wenn das Datenbanksystem erstellt ist, folgen Phasen der Wartung und Optimierung der physischen Ebene sowie die Anpassung an neue Systemplattformen. In der Regel übersteigen die anfallenden Kosten in dieser Phase die Entwurfskosten [SSH08, Seite 130].

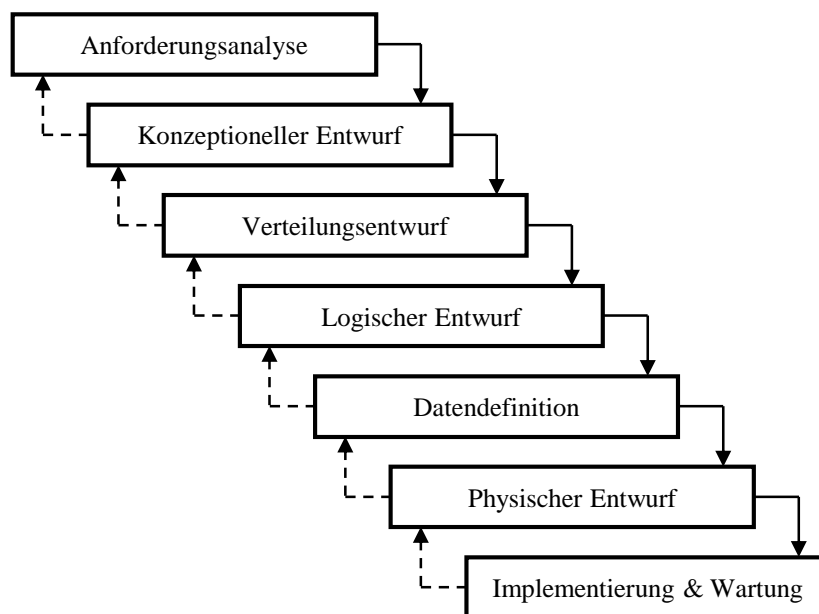


Abbildung 2-2: Phasen des Datenbankentwurfs [SSH08, Seite 124]

2.3 Die konzeptionelle Phase – Das ER-Modell

Das Entity-Relationship-Modell, das in den 70ern von Codd entwickelt wurde, ist ein abstraktes Modell, das einen Ausschnitt der realen Welt beschreibt. Es setzt sich aus Entities, Beziehungen und Attributen zusammen. Mit Hilfe dieses Modells kann die Struktur der Tabellen und deren Beziehungen untereinander in einer Datenbank entworfen und visualisiert werden.

Eine **Entity** beschreibt ein Objekt der realen Welt oder der Vorstellungswelt über das Informationen zu speichern sind, z.B. ein Kunde, ein Lieferant oder ein Arbeitsplan. Im ER-Modell wird eine Entity als Rechteck (siehe Abbildung 2-3) dargestellt. Zwischen zwei Entities kann es eine **Beziehung** (Raute) geben, z.B. Kunde bestellt Artikel. Eigenschaften einer Entity oder Beziehung werden als **Attribut** (Ellipse) bezeichnet. Ein Kunde kann die Attribute Name, Vorname und Geburtsdatum haben. Um die Schlüsseleigenschaft eines oder mehrerer Attribute anzuzeigen, wird es unterstrichen [SSH08, Seite 60].



Abbildung 2-3: Entity, Beziehung, Attribut

In einem ER-Modell wird die Anzahl der beteiligten Instanzen des jeweiligen Entity-Typs angegeben (Kardinalität) [SSH08, Seite 68]. Kardinalitäten werden wie folgt unterschieden:

- **1:1-Beziehung** – Jeder Entity aus der Menge E_1 ist genau eine Entity aus der Menge E_2 zugeordnet (siehe Abbildung 2-4) [SSH08, Seite 71].

Beispiel: Jede Maschine hat genau einen Wartungsplan. Ein Wartungsplan gilt für genau eine Maschine.

- **1:N- bzw. N:1-Beziehung** – Jeder Entity aus der Menge E_1 werden beliebig viele Entities aus der Menge E_2 zugeordnet (siehe Abbildung 2-4) [SSH08, Seite 71].

Beispiel: In einer Fabrikhalle stehen mehrere Maschinen. Eine Maschine steht in genau einer Fabrikhalle.

- **M:N-Beziehung** – Mehrere Entities aus der Menge E_1 können mehreren Entities aus der Menge E_2 zugeordnet werden (siehe Abbildung 2-4) [SSH08, Seite 71].

Beispiel: Eine Fabrik hat mehrere Zulieferer. Mehrere Zulieferer bedienen mehrere Fabriken.

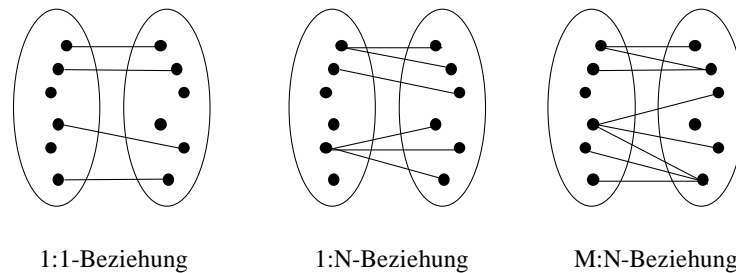


Abbildung 2-4: Beziehungen [SSH08, Seite 72 f.]

Es wird ein weiterer Beziehungstyp unterschieden, der im Datenbank-Konzept in Kapitel 4 von Bedeutung ist. Eine Entity kann in Beziehung mit sich selbst stehen (rekursive Beziehung) und dabei zwei unterschiedliche Rollen einnehmen. Z.B. setzt sich ein Produkt aus verschiedenen Baugruppen zusammen. Es nimmt dann die Rollen „geht ein in“ und „ist zusammengesetzt aus“ an. Rekursive Beziehungs-Typen werden eingesetzt, um Hierarchien abzubilden [Kud07, Seite 58].

Um Intervalle anzugeben, kann die (min, max)-Notation verwendet werden. Kardinalitätsangaben werden als Intervalle über einen minimalen und maximalen Wert realisiert. Das Minimum ist {0,1} und das Maximum {1,*}. Das Symbol „*“ bedeutet „unbegrenzt“. Es wird angegeben, an wie vielen Beziehungen eine Entity minimal und maximal beteiligt sein kann [Kud07, S.56].

3 Quovadis – Werkzeug zur Unterstützung der Konzeptplanung im Fabrikplanungsprozess

Quovadis ist ein Werkzeug zur Unterstützung der Fabrikplanung. Für das bessere Verständnis, welche Daten in der Datenbank gespeichert werden sollen, werden im Folgenden die Grundlagen der Fabrikplanung behandelt. Dabei wird detaillierter auf die Phase der Konzeptplanung im Fabrikplanungsprozess eingegangen sowie die Funktionsweise des Werkzeugs beschrieben und die Möglichkeiten für den Planer aufgezeigt. Im letzten Abschnitt dieses Kapitels werden die vorhandene Datenbankanbindung des Prototyps und die Struktur der Datenbank analysiert.

3.1 Fabrikplanung

Eine Fabrik ist ein industrieller Betrieb, in dem unter Einsatz von Produktionsfaktoren Erzeugnisse hergestellt werden. Durch Änderungen innerhalb der Fabrik oder neuer Anforderungen von außerhalb, kann eine Umstrukturierung der Fabrik erforderlich sein. Fabrikintern zählen dazu z.B. neue Produktionstechnologien oder organisatorische Änderungen. Aber auch die Änderung der Unternehmensstrategie oder die Entwicklung eines neuen Produktes sind Auslöser. Eine Fabrik muss sich jedoch auch an veränderte Marktsituationen, gesetzliche Vorgaben, individuelle Kundenwünsche oder Änderungen am Arbeitsmarkt anpassen. Die Fabrikplanung ist ein Prozess, der sieben aufeinander aufbauende Phasen umfasst, und mit Hilfe von Methoden und Werkzeugen eine Fabrik plant [VDI5200, Seite 3].

Bei der Planung einer Fabrik werden vier Grundfälle unterschieden: Neubau, Umgestaltung, Rückbau und Revitalisierung. Der **Neubau** ist der klassische Grundfall, bei dem auf der „grünen Wiese“ geplant wird und nur die Restriktionen des Geländes und der vorhandenen Infrastruktur berücksichtigt werden müssen. Das erfordert einen hohen zeitlich-inhaltlichen Planungsvorlauf, hat jedoch den Vorteil hoher Freiheitsgrade im Gestaltungsprozess. Ein weiterer Grundfall ist die **Umgestaltung** einer bestehenden Fabrik, welche den größten Anteil der Fabrikplanungsaufgaben umfasst. Sie ist oft eine betriebliche Daueraufgabe, da die Fabrik fortlaufend an Produktionsprogrammveränderungen angepasst werden soll. Dabei müssen die vorhandenen Produktionsfaktoren sowie der laufende Fabrikbetrieb beachtet werden. Ein **Rückbau** kann die Folge von Umsatzrückgang oder der Auslagerung von Produktionsstufen sein. Weiterhin müssen die Kapazitäten und Strukturen der Produktions- und Nebenbereiche angepasst werden. Die

Änderung des Produktionsprogramms kann eine erforderliche Maßnahme sein. Durch **Revitalisierung** wird ein Standort für eine industrielle Neunutzung vorbereitet. Es werden z.B. Gebäude neu gestaltet sowie Flächen- und Raumstrukturen saniert [VDI5200, Seite 3; Gru06, Seite 14 ff.].

3.1.1 Phasen der Fabrikplanung

Zunächst folgen Erklärungen von Begriffen, die im weiteren Verlauf der Arbeit von Bedeutung sind. Danach werden die Phasen der Fabrikplanung erläutert.

Bei der Fabrikplanung werden fünf Planungsebenen differenziert, die in Abbildung 3-1 veranschaulicht sind. Die kleinste Ebene stellt der **Arbeitsplatz** dar. Das ist eine Stelle in der Fabrik, die von einem oder mehreren Arbeitnehmern verwendet wird und aus Betriebsmitteln und Nebeneinrichtungen besteht. Mehrere Arbeitsplätze werden in einem **Segment** zu einem geschlossenen Produktionsbereich zusammengefasst. Eine Ebene über dem Segment ist das **Gebäude**, welches einen „architektonisch-räumlich geschlossenen Produktionsbereich“ darstellt und aus mehreren Segmenten bestehen kann. Ein **Werk** umfasst mehrere Gebäude mit Wegen, Außenanlagen und Anbindung an die externe Infrastruktur. Die größte Ebene ist das Produktionsnetz, welches Werke verschiedener Standorte miteinander verbindet [VDI5200, Seite 4]. In dieser Arbeit werden alle Ebenen, außer dem Produktionsnetz, von Bedeutung sein.

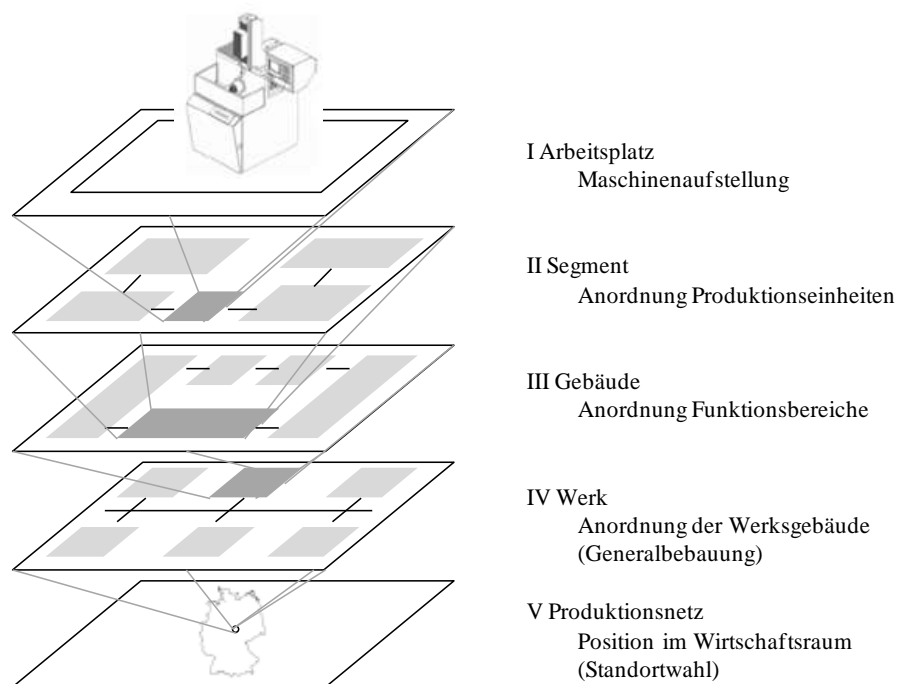


Abbildung 3-1: Planungsebenen [Sch10, Seite 12]

Die Bezeichnungen bzgl. der Ebenen aus den Quellen [VDI5200] und [Gru06] unterscheiden sich jedoch. Tabelle 1 stellt die Unterschiede gegenüber. Ab Kapitel 3.1.2 werden die Begriffe von [Gru06] verwendet.

Tabelle 1: Gegenüberstellung der Bezeichnungen

Prototyp (Grundig)	Erweiterung (VDI 5200)
Betrieb	Gebäude
Funktionsbereich	Segment
Betriebsmittel	Arbeitsplatz

In der Fachliteratur werden verschiedene Phasenmodelle für die Fabrikplanung beschrieben. Um den Fabrikplanungsprozess zu vereinheitlichen, wurde die VDI-Richtlinie 5200: Fabrikplanung entwickelt. Diese Arbeit orientiert sich an dieser Richtlinie.

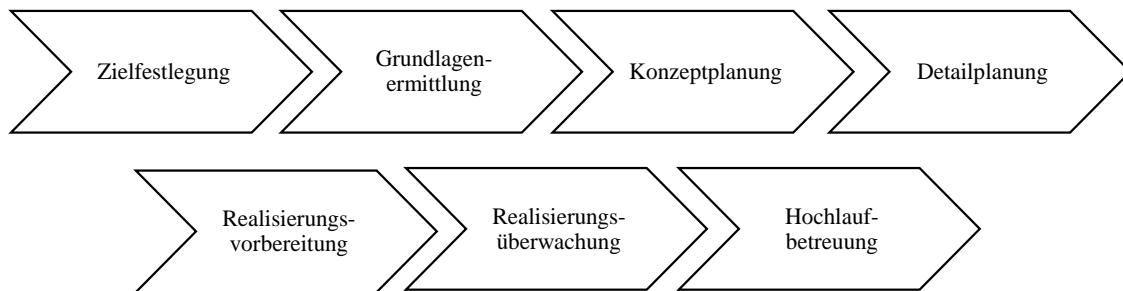


Abbildung 3-2: Fabrikplanungsprozess

Als erstes wird in der **Zielfestlegung** die fabrikplanerische Aufgabe gestellt und das Planungsprojekt strukturiert. Das Ergebnis dieser Phase sind die festgelegten Fabrikziele mit ihren gewichteten Bewertungskriterien, die Ziele des Projektes und der Projektplan. In der **Grundlagenermittlung** werden die für die Planung benötigten Daten und Informationen gesammelt oder erstellt. Diese umfassen Daten zu Mengengerüsten, Betriebsmitteln, Produktionsabläufen, Gebäuden und Grundstücken. Des Weiteren wird aus der Sicht der Fabrikplanung die Aufgabenstellung des Projektes genauer formuliert. In der **Konzeptplanung** wird die Fabrik in ihrer Gesamtheit entworfen und ein umsetzbares Fabrikkonzept entwickelt. Das Ergebnis ist ein Groblayout und Gebäudevorentwurf. In der darauf folgenden **Detailplanung** werden die ausgewählte Variante des Fabrikkonzepts aus der Konzeptplanung verfeinert und die Elemente der Fabrik detailliert beschrieben und grafisch dargestellt. Das Ergebnis ist das Feinlayout, das im Gegensatz zu dem Groblayout die Ausführungspläne der Gebäude enthält. In der **Realisierungsvorbereitung** wird basierend auf den Leistungsbeschreibungen der Fabrikelemente die

Vergabe an die Lieferanten organisiert und die Umsetzung der Fabrik geplant. Im Rahmen der **Realisierungsüberwachung** wird die ordnungsgemäße Ausführung der Gebäude mit Außenanlagen und mit den Betriebsmitteln sichergestellt und dokumentiert. Das Ergebnis ist eine erstellte Fabrik. In der Phase **Hochlaufbetreuung** wird die Fabrik in Betrieb genommen und anhand der Fabrikziele bewertet. Ziel ist die Erreichung der in der Phase Zielfestlegung definierten Fabrikziele [VDI5200, Seite 5 ff.].

In Kapitel 3.1.2 wird auf die Phase der Konzeptplanung näher eingegangen. Die anderen Phasen werden nicht weiter betrachtet, da Quovadis zunächst die Phase der Konzeptplanung umsetzt.

3.1.2 Konzeptplanung

Die Konzeptplanung besteht aus vier Schritten, die nacheinander abgearbeitet werden und ggf. wiederholt werden können (siehe Abbildung 3-3) [VDI 5200, Seite 7].



Abbildung 3-3: Konzeptplanung

In der **Strukturplanung** werden die erforderlichen funktionalen und organisatorischen Einheiten (Bereiche, Ausrüstungen) festgelegt. Die Produktion wird nach Kriterien strukturiert. Bereiche können produkt-, produktstruktur-, personal-/tätigkeits-, materialfluss-, betriebsmittel- oder werkstofforientiert sein. Die somit ermittelte Produktionsstruktur wird in mehreren Varianten gestaltet und bewertet, um dann zu einem idealen Funktionsschema zusammengeführt zu werden. Das Funktionsschema dient der übersichtlichen Darstellung der Produktionsabläufe und als Ausgangsbasis für die weitere Planung. Daraus gehen alle Produktionsprozessabfolgen, aber nicht die räumliche Anordnung, hervor [VDI5200, Seite 7].

In der darauf folgenden **Dimensionierung** wird der Flächenbedarf der Einheiten ermittelt. Basierend auf den Planungsdaten werden die Art und Menge der Betriebsmittel festgelegt. Als Betriebsmittel „werden allgemein Ausrüstungen (Maschinen), Anlagen, Vorrichtungen, Messmittel, Werkzeuge u.a. bezeichnet (technische Arbeitsmittel)“ [Gru06, Seite 73]. Aus der bestimmten Anzahl der Betriebsmittel leitet sich der Personal-, Flächen- und Medienbedarf ab. Durch z.B. verschiedene Technologien entstehen

Varianten. Auf der Grundlage des idealen Funktionsschemas wird ein Materialflusskonzept entwickelt. Im Anschluss daran können die Mengen und Frequenzen der Materialflüsse zwischen den funktionalen Einheiten und die zugehörigen Flächenbedarfe ermittelt werden [VDI5200, Seite 7].

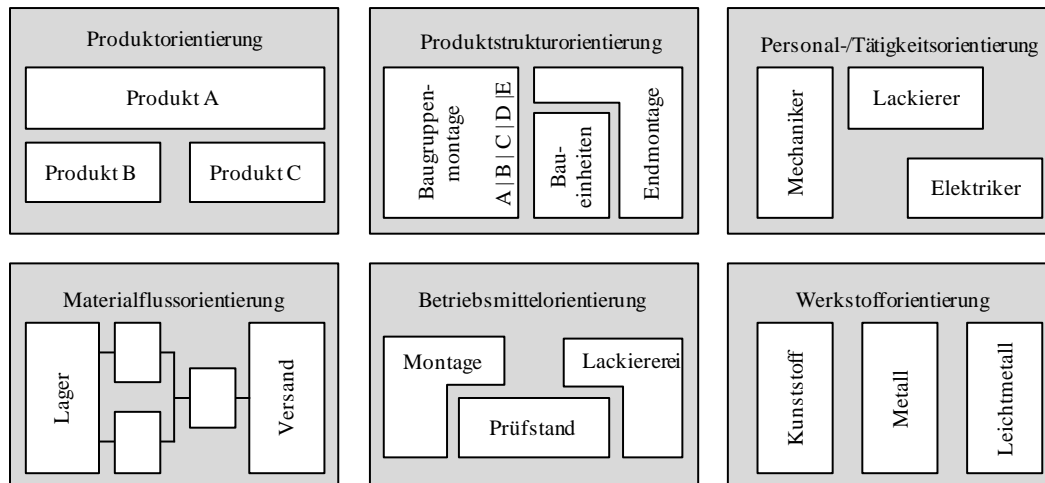


Abbildung 3-4: Strukturierungsprinzipien [Poe08, Seite 9]

Die **Idealplanung** dient zur Absicherung der Fabrikziele. Die zuvor bestimmten Einheiten aus der Phase der Dimensionierung werden, ohne Restriktionen zu beachten, räumlich ideal angeordnet. Bei der Erstellung von Layoutvarianten sollen die Anordnungen den qualitativen Fabrikzielen entsprechen. Restriktionen, die auf das Layout bezogen sind, wie z.B. wirtschaftliche und gesetzliche Beschränkungen oder anlagentechnische Anforderungen, bleiben in dieser Phase unberücksichtigt [VDI5200, Seite 7].

In der **Realplanung** werden im Gegensatz zu der Idealplanung bei der Erzeugung und Bewertung von Layoutvarianten alle Restriktionen einbezogen. Ein eindeutig optimales Konzept kann nicht ermittelt werden, nur Lösungsvarianten, die die Planungsaufgabe bestmöglich erfüllen [VDI5200, Seite 7 f.]. Abschließend werden die Lösungsvarianten sowie die Idealvariante nach gewichteten Bewertungskriterien monetär, quantitativ und qualitativ bewertet, um daraus eine Vorzugsvariante auswählen zu können. Das Ergebnis ist ein realisierbares Fabrikkonzept [VDI5200, Seite 8].

3.2 Funktionsweise des Werkzeugs

Am Fraunhofer IFF wird gegenwärtig ein Werkzeug entwickelt, das auf fundierte Fabrikplanung ausgelegt ist, wobei Aspekte der Standortplanung bisher ausgenommen sind.

Es werden die Fälle der Neu- und Umplanung betrachtet. Im Gegensatz zu dem Prototyp, der für die Automobilbranche entwickelt wurde, soll die Erweiterung branchenübergreifend eingesetzt werden. Quovadis beschränkt sich auf die Phase der Konzeptplanung im Fabrikplanungsprozess. Abbildung 3-5 zeigt den Aufbau von Quovadis.

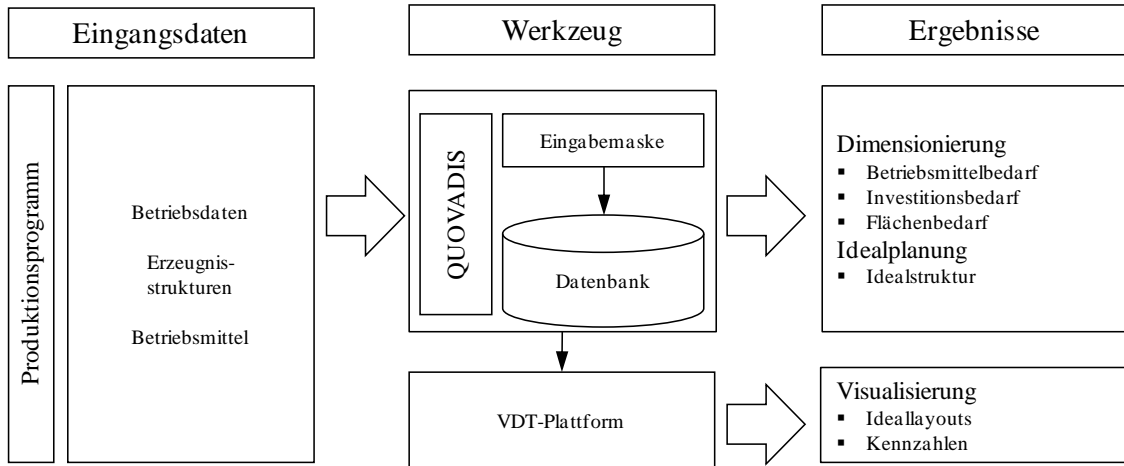


Abbildung 3-5: Funktionsweise von Quovadis [Sch10, Seite 9]

Bei Planungsbeginn sind zunächst alle planungsrelevanten Daten zu erheben und als Datenbasis im Werkzeug zu hinterlegen. Sie leiten sich aus dem Produktionsprogramm ab und umfassen Angaben zum Betrieb, Funktionsbereichen, Erzeugnissen, Erzeugniselementen und Prozessen. Abbildung 3-6 gibt eine Übersicht über die zu erfassenden Daten.

Betrieb	Funktionsbereich	Erzeugnis	Erzeugniselement	Prozess	Betriebsmittel	Technologie
Name	Betrieb	Betrieb	Betrieb	Betrieb	Name	Name
Arbeitstage	Name	Name	Name	Funktionsbereich	Investitionskosten	Technisches Kriterium
Tagesarbeitszeit	Fertigungsart	Fertigungsart	Erstellungsart	Technologie	Betriebskosten	
Fertigungsart	Tagesarbeitszeit	Produktionsmenge	Ausprägung	Betriebsmittel	Abmessung außen	
	Flächenzuschlagsfaktor	Tagesarbeitszeit		Name	Tagesarbeitszeit	
	Kostenzuschlagsfaktor			Bearbeitungsintensität	Stationen	
				Bearbeitungszeit	Einschränkungsfaktor	

Abbildung 3-6: Planungsrelevante Daten

Der Planer gibt die Erzeugnisstruktur für jedes zu produzierende Erzeugnis an, woraus die Baustruktur hervorgeht. In Abbildung 3-7 ist die Erzeugnisstruktur für das Produkt „Fahrrad“ dargestellt. Daraus geht hervor, aus welchen Baugruppen und Einzelteilen

(Erzeugniselementen) ein Fahrrad besteht und die Anzahl der benötigten Teile. Bspw. besteht die Baugruppe „Rahmen komplett“ aus einem Vorderrad, einer Gabel, zwei Schutzblechen, einer Kette, zwei Pedalen, einem Hinterrad und dem Rahmen selbst.

In der Erzeugnisstruktur wird jedem Element genau ein Prozess zugeordnet, welcher von einem Betriebsmittel ausgeführt wird. Daraus ergeben sich Vorgänger-Nachfolger-Beziehungen, welche die Abfolge der Produktion (Arbeitsplan) sowie die Kopplung der Betriebsmittel und Funktionsbereiche bestimmen.

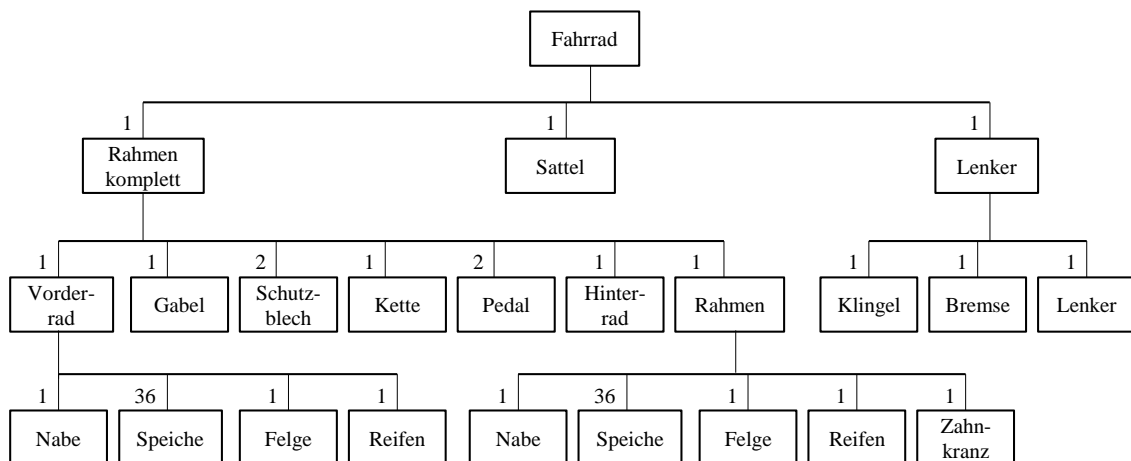


Abbildung 3-7: Erzeugnisstruktur für ein Fahrrad

Das Werkzeug umfasst eine Betriebsmittelbibliothek über die Betriebsmittel, die durch Technologien und ihre Abmessungen definiert sind, den Prozessen zugeordnet werden [Sch10, Seite 5]. Durch Filterung anhand der Technologie soll die Auswahl der Betriebsmittel erleichtert werden.

Aus der anzugebenden Maschinengrundfläche und mittels eines Zuschlagsfaktors wird die Arbeitsplatzfläche eines Betriebsmittels berechnet. Mit dem Zuschlagsfaktor, der vom Planer gewählt werden kann, können zusätzliche Flächenanteile, z.B. Bedien-, Wartungs- und Schutzflächen, berücksichtigt werden. Alternativ kann die Arbeitsplatzfläche eines Betriebsmittels nach der Formel 3.1 von [Rock82, Seite 55] durch das Werkzeug berechnet werden:

$$(3.1) \quad A_{A_i} = 4,6 \cdot A_{G_i} - 0,16 \cdot A_{G_i}^2 + 0,0018 \cdot A_{G_i}^3 \quad [\text{m}^2]$$

A_{A_i} Arbeitsplatzfläche des Betriebsmittel i inkl. Bedien- und Bereitstellflächen sowie Flächen für Reparatur, Wartung und Abfallprodukte [m^2]

A_{G_i} Grundfläche des Betriebsmittels i [m^2]

f_G Zuschlagsfaktor für Bereitstellung, Bedienung, Wartung sowie Ver- und Entsorgung am Arbeitsplatz

Die Maschinenarbeitsplatzfläche stellt die kleinste Fläche dar, aus der alle weiteren Flächen der Fabrik abgeleitet werden. Die Bereichsbildung wird manuell vom Planer vorgenommen, indem er die Funktionsbereiche festlegt und die Betriebsmittel zuordnet. Die Einheiten werden daraufhin dimensioniert und angeordnet [Sch10, Seite 5]. Dazu zählen die Anzahl der Betriebsmittel, benötigte Flächen und entstehende Investitionskosten. Durch Stücklistenauflösung¹ werden aus den zu produzierenden Stückzahlen der Erzeugnisse und deren Erzeugnisstruktur die Stückzahlen für die Erzeugniselemente bestimmt, welche die Basis für die Berechnung des Materialflusses darstellen.

Um die Anzahl der Betriebsmittel zu errechnen, erfolgt zunächst ein Kapazitätsabgleich, bei dem die benötigte Bearbeitungszeit dem verfügbaren Zeitangebot gegenübergestellt wird (Formel 3.2). Das Zeitangebot ergibt sich aus der Anzahl der Arbeitstage des Betriebs und der Tagesarbeitszeit der Betriebsmittel. Die benötigte Bearbeitungszeit wird aus der Prozesszeit und Stückzahl des zu fertigenden Erzeugnisses berechnet. Mit dem Einschränkungsfaktor können zusätzliche Zeiten wie z.B. Rüstzeiten² einbezogen werden [Sch10, Seite 6].

$$(3.2) \quad BM_i = \frac{\sum_{p=1}^P (m_{eip} * t_{eip})}{t_b * t_i} * \eta_i$$

BM_i Anzahl der erforderlichen Betriebsmittel i [-]

m_{eip} Anzahl des Prozesses e auf dem Betriebsmittel i zur Erstellung von Erzeugnis p [1/Jahr]

t_{eip} Bearbeitungszeit je Prozess e auf dem Betriebsmittel i zur Erstellung von Erzeugnis p [min]

t_b Betriebsarbeitstage für den Betrieb b [Tage/Jahr]

t_i Tagesarbeitszeit für das Betriebsmittel i [min/Tag]

η_i Einschränkungsfaktor für das Betriebsmittel i [-]

Aus der ermittelten Anzahl der Betriebsmittel und den hinterlegten Flächen können die Flächen der Funktionsbereiche berechnet und als deren Summe die gesamte Produkti-

¹ „Eine Stückliste [...] ist eine tabellarische Darstellung der Erzeugnisstruktur [...]. [HL99, Seite 425]

² „Technologische Vorbereitungs- und Abschlusszeit für einen Arbeitsgang zur Realisierung eines Loses oder Auftrages in einem Arbeitssystem [...]“. [Hel10, Seite 1397]

onsfläche bestimmt werden. Weitere Flächenbedarfe für Lager- und Verkehrsflächen werden über den Zuschlagsfaktor einbezogen (Formel 3.3) [Sch10, Seite 6].

$$(3.3) \quad A_{P_f} = f_A \cdot \sum_{i=1}^l A_{A_{if}}$$

A_{P_f} Produktionsfläche des Funktionsbereiches f [m²]

f_A Zuschlagsfaktor für Lager, Transport, Qualitätssicherung, Ver- und Entsorgung sowie Produktionssteuerung

$A_{A_{if}}$ Arbeitsplatzfläche des Betriebsmittels i im Funktionsbereich f [m²]

Die Investitionskosten der benötigten Betriebsmittel und die Kosten für Flächen werden mittels eines Flächenkostenfaktors errechnet. Die Materialflüsse, welche sich aus den Vorgänger-Nachfolger-Beziehungen der Erzeugnisstruktur und den Prozessen ableiten, werden auf den Ebenen des Betriebs und der Funktionsbereiche berechnet und in Form von Matrizen zusammengefasst. Aus ihnen gehen die Von-Nach-Flussbeziehungen in tabellarischer Form hervor. Anhand der Stückzahlen werden im Prototyp die Materialflüsse ermittelt.

Das Werkzeug dimensioniert in synthetischer Richtung, das heißt, dass von der kleinsten Struktureinheit, z.B. einem Arbeitsplatz, auf das Gesamtsystem geschlossen wird [KSG84, Seite 10; Sch10, Seite 6 f.].

Um ausgewählte und dimensionierte Betriebsmittel anzuordnen, wird zunächst die Fertigungsform (Punkt-, Werkstatt-, Nest-, Reihen- oder Fließfertigung)³ bestimmt. Der Kooperationsgrad⁴ (Formel 3.4) [Sch70, Seite 19] unterstützt die Wahl der Fertigungsform, welche jeweils einem Strukturtypen entspricht, z.B. entspricht der Werkstattfertigung die Werkstattstruktur. Der Strukturtyp kann als räumliches Anordnungsprinzip der Fertigungsform angesehen werden, anhand dessen die Methode der Anordnung der Betriebsmittel ausgewählt wird [Sch10, Seite 7 f.].

$$(3.4) \quad \kappa_f = \frac{\sum_{i=1}^m k_{if}}{m_f} [-]$$

κ_f Kooperationsgrad im betrachteten Funktionsbereich f [-]

³ Nachzulesen in [Gru06, Seite 115 ff.]

⁴ „Mit dem Kooperationsgrad K wird die durchschnittliche Anzahl von Arbeitsplätzen bezeichnet, mit denen ein Arbeitsplatz (...) aufgrund des Teiledurchlaufes (Materialfluss) unmittelbar verbunden ist.“

- k_i Anzahl der Betriebsmittel mit denen das Betriebsmittel i im Funktionsbereich f unmittelbar in Verbindung steht [-]
- m Anzahl der Maschinen im betrachteten Funktionsbereich

Die Ergebnisse der Berechnungen werden in der Datenbank gespeichert und an ein externes Werkzeug, die Virtual Development and Training-Plattform⁵ (VDT-Plattform) übergeben, um die Anordnungen darzustellen und zu visualisieren. Dazu werden sie in eine XML-Datei umgewandelt. Zuvor generierte Anordnungen werden als Ausgangsszenario gespeichert. In der VDT-Plattform können Änderungen vorgenommen und Varianten erstellt werden, die dann als alternative Szenarien hinterlegt werden. Die Szenarien werden mit Hilfe von Kennzahlen, wie der Auslastung je Betriebsmitteltyp, durchschnittliche Auslastung aller Betriebsmittel sowie Investitionskosten und Flächenausnutzung, verglichen [Sch10, Seite 8 f.].

3.3 Analyse der vorhandenen Datenbankanbindung und der Struktur der Datenbank

Nachdem die Funktionsweise des Werkzeugs erläutert wurde, soll eine Analyse der vorhandenen Datenbank folgen. Das ER-Modell ist in Abbildung 3-8 dargestellt.

Die Abbildung zeigt, dass ein Betrieb in mehrere Funktionsbereiche gegliedert ist, die jeweils mehrere Prozesse ausführen. Jeder Prozess verwendet genau eine Technologie und ein Betriebsmittel. Ein Betrieb produziert außerdem Erzeugniselemente, die wiederum mehreren Erzeugnissen zugeordnet sind. In der Baustruktur werden die Baugruppen und Einzelteile, die Erzeugniselemente darstellen, erfasst und ihnen jeweils Prozess zugeordnet. Ein und dasselbe Erzeugniselement kann in mehreren Baugruppen auftreten. Im Szenario werden die verschiedenen Baugruppen erfasst und einem Betrieb zugeordnet.

Die Beziehungen zwischen Prozess, Betriebsmittel und Technologie im Prototyp sind nicht vollständig, da die 1:N-Beziehung „Betriebsmittel verwendet Technologie“ fehlt. Außerdem erfolgt die Planung auf Betriebsebene, in der Erweiterung soll jedoch ein Werk, das aus mehreren Betrieben bestehen kann, betrachtet werden. Das Erzeugnis wird dann auch nicht dem Erzeugniselement zugeordnet, sondern dem Werk. Auch ist

⁵ Ist ein „am Fraunhofer IFF entwickeltes interaktives Visualisierungssystem (...) mit dem Produkte und Prozesse in der virtuellen Welt“ visualisiert und getestet werden können. www.iff.fraunhofer.de

die 1:N-Beziehung zwischen Erzeugnis und Erzeugniselement fehlerhaft. Es ist richtig, dass ein Erzeugniselement mehreren Erzeugnissen zugeordnet werden kann, jedoch besteht ein Erzeugnis aus mehreren Elementen. Diese M:N-Beziehung soll in der Erweiterung berücksichtigt werden.

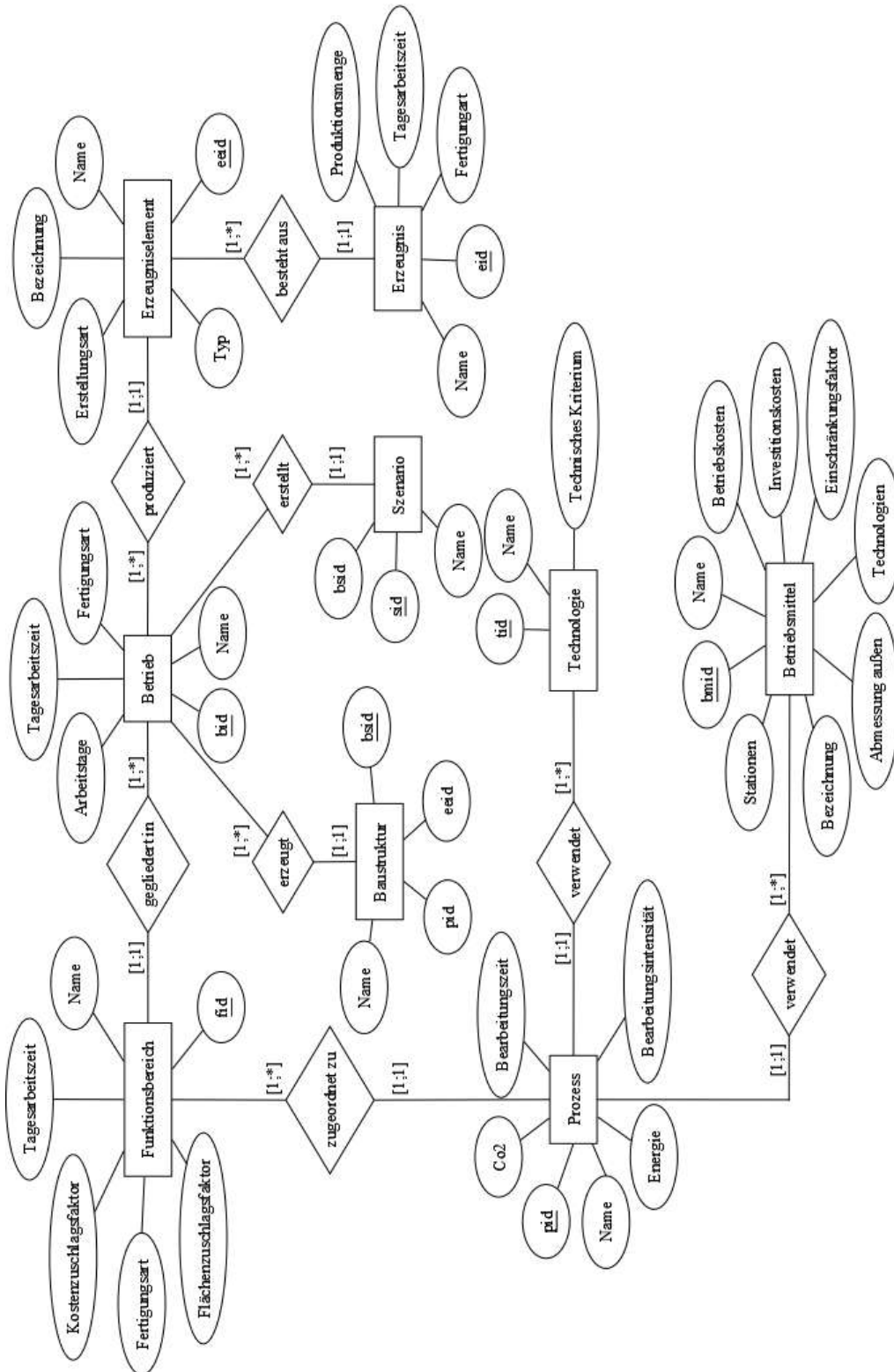


Abbildung 3-8: ER-Modell des Prototyps

Durch die Zuordnung der Funktionsbereiche und der Erzeugniselemente zum Betrieb, eignet sich der Prototyp zur Segmentierung anhand funktioneller Kriterien. Jedoch soll es auch möglich sein, dass ein Funktionsbereich genau ein Erzeugnis fertigt (Produktorientierung) [Sch10, Seite 24 f.].

Im ER-Modell des Prototyps werden zwar einem Erzeugniselement mehrere Prozesse zugeordnet, jedoch geht dabei nicht die Reihenfolge dieser hervor. Die Angabe von Arbeitsplänen ist somit nicht möglich. Außerdem fehlt die Information auf welchen Betriebsmitteln das Erzeugniselement bearbeitet wird.

Im Prototyp Quovadis dient die Entity Szenario dem Erfassen von Baustrukturen. In der Erweiterung werden jedoch die unterschiedlichen möglichen Anordnungen von Funktionsbereichen im Betrieb gespeichert. Weiterhin ist es nicht möglich Dokumente im Prototyp zu speichern und Flächen anzugeben. Im Prototyp wurden ein SQL-Datenbanksystem und ein Microsoft-SQL-Server 2005 als DBMS verwendet, was weiterhin beibehalten wird.

4 Datenbank-Konzept

Nachdem die Grundlagen von Datenbanken und der Fabrikplanung mit Hilfe des Quovadis Prototyps in den vorherigen Kapiteln behandelt wurden, wird im Folgenden das Datenbankkonzept für die Erweiterung von Quovadis erarbeitet. Dieses Kapitel folgt dem Modell für den Datenbankentwurf aus Kapitel 2. Zunächst werden in der Anforderungsanalyse die notwendigen Anforderungen an die Datenbank abgeleitet und daraus das konzeptionelle Schema erstellt, welches im logischen Entwurf in ein relationales Modell transformiert wird. Die Phasen Verteilungsentwurf, Datendefinition, physischer Entwurf sowie Implementierung und Wartung sind nicht Bestandteil dieser Arbeit, da die Aufgabenstellung sich auf die Erarbeitung eines Konzeptes beschränkt.

4.1 Ableitung notwendiger Anforderungen - Anforderungsanalyse

Die Anforderungsanalyse umfasst die Sammlung aller Informationen. Hier wird angegeben, was und wie Daten gespeichert werden. Die folgenden Anforderungen werden durch Analyse der vorhandenen konzeptionellen Arbeiten zusammengestellt. Diese werden nach [Voss08, Seite 54] in Informations- und Verarbeitungsanforderungen eingeteilt.

4.1.1 Informationsanforderungen

Die Erweiterung behält die Begriffe des Prototyps bei. Es werden die in Kapitel 3 eingeführten Bezeichnungen von [Gru06] verwendet. Im Folgenden werden die zu speichernden Entities mit ihren Attributen angegeben und beschrieben. Dabei wird die Erweiterung dem Prototyp gegenübergestellt, wodurch ersichtlich wird, welche Attribute übernommen und welche hinzugefügt werden. Zu jedem Attribut werden der Speicherbedarf in Klammern und der Datentyp, der den Inhalt einer Variablen bezeichnet, angegeben. Tabelle 2 zeigt die verschiedenen Datentypen, die Microsoft-SQL-Server 2005 anbietet. Ein Attribut, das als Primärschlüssel dienen kann, ist unterstrichen. Zusätzlich werden in der letzten Spalte der Tabellen Bemerkungen zu Wertebereichen, Einheiten oder der möglichen Auswahl gemacht.

Im Fall der Umplanung einer Fabrik, sind bereits Gebäude, Funktionsbereiche und Betriebsmittel usw. vorhanden. Informationen diesbezüglich werden als Grunddaten in der Datenbank gespeichert und im Folgenden vorgestellt. Danach wird auf die Ergebnisdar-

ten eingegangen und deren Beziehungen untereinander sowie zu den Grunddaten betrachtet.

Tabelle 2: Datentypen

Datentyp	Bedeutung
int	Ganzzahl
float	Gleitkommazahlen
text	Text
varchar	Zeichenketten variabler Länge
char	Zeichenketten fester Länge
blob	Große binäre Objekte
clob	Lange Zeichenketten

Der Prototyp hat die Fabrik auf Betriebsebene betrachtet, die Erweiterung dagegen soll die Fabrik auf Werksebene sehen. Deshalb wird die Entity **Werk** eingeführt, die den Namen diesbezüglich speichert (siehe Tabelle 3).

Tabelle 3: Entity Werk

Prototyp	Erweiterung		
-	Werk		
Attribut	Attribut	Datentyp	Bemerkung
	<u>Werk</u> (20)	char	

Ein Werk kann aus mehreren **Betrieben** bestehen. Tabelle 4 zeigt, dass aus dem Prototyp alle Attribute bis auf den Schlüssel übernommen werden. Die Fertigungsart und die Tagesarbeitszeit werden zunächst für den gesamten Betrieb angegeben und automatisch für die niedrigeren Ebenen Funktionsbereich, Erzeugnis und Betriebsmittel übernommen, um ein späteres mehrmaliges Abfragen dieser Daten zu vermeiden. Sie können jedoch jederzeit vom Planer geändert werden und nehmen dabei keinen Einfluss auf die jeweils übergeordnete Ebene. Die Tagesarbeitszeit muss spätestens auf der Betriebsmittelebene angegeben werden, da diese für die Berechnungen der Kapazitäten nötig ist. Durch das mehrmalige Speichern dieser Attribute in verschiedenen Entities entstehen auf der einen Seite Redundanzen, aber auf der anderen Seite muss die Angabe spezifiziert werden, da es zwischen Erzeugnissen, Funktionsbereichen und Betriebsmitteln zu Überschneidungen kommen kann, da ein Erzeugnis während der Fertigung verschiedene

Funktionsbereiche durchlaufen kann [Poe08, Seite 55 f.]. Der Kostenzuschlagsfaktor dient zur Berücksichtigung der Kosten für zusätzliche Flächen. Über den Namen wird der Betrieb eindeutig identifiziert.

Tabelle 4: Entity Betrieb

Prototyp	Erweiterung		
Betrieb	Betrieb		
Attribut	Attribut	Datentyp	Bemerkung
<u>bid</u>			
Name	<u>Betrieb</u> (20)	char	
Tagesarbeitszeit	Tagesarbeitszeit	float	Stunden [0;24]
Arbeitstage	Arbeitstage	int	Tage [0;365]
Fertigungsart	Fertigungsart (20)	char	Einzelfertigung, Klein-/Mittel-/Großserie, Massenfertigung
	Kostenzuschlagsfaktor	float	

Ein Betrieb, also das Gebäude im engeren Sinn, hat mehrere **Etagen**, die eine Länge und Breite aufweisen. Diese Werte werden gespeichert, damit später die Bereiche auf diesen Flächen angeordnet werden können (Tabelle 5). Im Prototyp wurden bisher keine Flächen und somit auch nicht die Anordnung der Funktionsbereiche berücksichtigt.

Tabelle 5: Entity Etage

Prototyp	Erweiterung		
-	Etage		
Attribut	Attribut	Datentyp	Bemerkung
	<u>Betrieb</u> (20)	char	
	<u>Etage</u>	int	
	Länge	float	Meter
	Breite	float	Meter

Weiterhin ist ein Betrieb in mehrere **Funktionsbereiche** unterteilt. In Tabelle 6 ist zu erkennen, dass die Attribute Name, Tagesarbeitszeit, Fertigungsart, Kosten- und Flächenzuschlagsfaktor aus dem Prototyp beibehalten werden. Da die Fertigungsart und Tagesarbeitszeit automatisch vom Betrieb übernommen werden, besteht hier die Möglichkeit die Angaben anzupassen. Der Flächenzuschlagsfaktor dient dazu, zusätzliche

Flächen für Wege, Lager und Transport zu berücksichtigen. Der Kostenzuschlagsfaktor gibt den Kostensatz für die Errichtung oder den Umbau von Flächen an.

Tabelle 6: Entity Funktionsbereich

Prototyp	Erweiterung		
Funktionsbereich	Funktionsbereich		
Attribut	Attribut	Datentyp	Bemerkung
<u>fid</u>			
Name	<u>Funktionsbereich</u> (20)	char	
Tagesarbeitszeit	Tagesarbeitszeit	float	Stunden [0;24]
Fertigungsart	Fertigungsart (20)	char	Einzelfertigung, Klein-/Mittel-/Großserie, Massenfertigung
Flächenzuschlagsfaktor	Flächenzuschlagsfaktor	float	
Kostenzuschlagsfaktor	Kostenzuschlagsfaktor	float	

Als **Betriebsmittel** werden Maschinen zur Fertigung von Erzeugnissen bezeichnet. Kerninformationen der zur Auswahl stehenden Betriebsmittel werden in der Entity Betriebsmittel hinterlegt. Tabelle 7 zeigt, dass die Attribute Name, Investitionskosten, Betriebskosten, Bezeichnung und Tagesarbeitszeit aus dem Prototyp übernommen werden. Die Tagesarbeitszeit sowie die jetzt zusätzliche Fertigungsart werden automatisch vom Betrieb übernommen und können hier angepasst werden. Mit Hilfe der Angaben zur Grundfläche des Betriebsmittels und dem Zuschlagsfaktor kann die Arbeitsplatzfläche des Betriebsmittels errechnet werden. Die Angabe der Länge und Breite ist für die Bestimmung der Flächen der Funktionsbereiche von Bedeutung. Die Investitionskosten bezeichnen die Anschaffungskosten für das Betriebsmittel. Mit der Bezeichnung kann die Maschine genauer beschrieben werden.

Wenn benötigte Betriebsmittel nicht vorhanden sind, besteht für den Planer die Möglichkeit neue Einträge anzulegen. Zunächst ist die Anzahl der Einträge gering. Sie wächst mit Erweiterung des Werkzeugs.

Die **Technologie** bezeichnet Fertigungsverfahren, also Prozesse zur Herstellung von Erzeugnissen. Die Verfahren werden nach der DIN 8580 in die Hauptgruppen Urformen, Umformen, Trennen, Fügen, Beschichten und Stoffeigenschaften ändern unterteilt [DIN8580, Seite 2]. Die Hauptgruppen sind wiederum in Gruppen gegliedert.

Tabelle 8 zeigt, dass der Name übernommen wird. Das technische Kriterium wurde durch eine Beschreibung ersetzt. Ein technisches Kriterium kann somit immer noch angegeben werden.

Zusätzlich soll nun die Obertechnologie und die Ausprägung angegeben werden. Zunächst sind nur ausgewählte Technologien vorhanden. Durch Weiterentwicklungen und Ergänzungen in Planungsprojekten wird die Sammlung erweitert. Die Auswahl der Technologie wird auf die essentiellen Verfahren im Bereich der industriellen Fertigung beschränkt und ist an der DIN 8580 orientiert. Der Planer wählt für ein Planungsobjekt die entsprechende Technologie aus, wodurch die Auswahl der in Frage kommenden Betriebsmittel sich einschränkt und ein Verfahren aus der gewählten Hauptgruppe ausgewählt werden kann. Es sollen auch Technologien beachtet werden können, die in der DIN 8580 noch keine Berücksichtigung gefunden haben [Poe08, Seite XIV].

Tabelle 7: Entity Betriebsmittel

Prototyp	Erweiterung		
Betriebsmittel	Betriebsmittel		
Attribut	Attribut	Datentyp	Bemerkung
<u>bmid</u>			
Name	<u>Betriebsmittel</u> (20)	char	
Investitionskosten	Investitionskosten	float	Euro
Abmessung außen			
Betriebskosten	Betriebskosten	float	Euro
Bezeichnung	Bezeichnung (30)	char	
Tagesarbeitszeit	Tagesarbeitszeit	float	Stunden [0;24]
	Fertigungsart (20)	char	Einzelfertigung, Klein-/Mittel-/Großserie, Massenfertigung
	Grundfläche	float	Meter ²
	Zuschlagsfaktor	float	
	Länge	float	Meter
	Breite	float	Meter
Stationen			
Technologien			
Einschränkungsfaktor			

Tabelle 8: Entity Technologie

Prototyp	Erweiterung		
Technologie	Technologie		
Attribut	Attribut	Datentyp	Bemerkung
<u>tid</u>			
Name	<u>Technologie</u> (20)	char	Auswahl nach der DIN 8580
Technisches Kriterium	Beschreibung (50)	float	
	Obertechnologie (20)	char	
	Ausprägung (30)	float	

Ein **Erzeugnis** bezeichnet einen hergestellten Gegenstand und kann aus Baugruppen, Einzelteilen und Rohmaterial bestehen. Die Attribute Fertigungsart, Tagesarbeitszeit und Produktionsmenge werden aus dem Prototyp übernommen. Hier besteht wieder die Möglichkeit die Tagesarbeitszeit und die Fertigungsart anzupassen. Jedes Erzeugnis ist durch eine eindeutige Materialnummer gekennzeichnet. Die zu fertigende Menge pro Jahr wird über die Produktionsmenge angegeben (Tabelle 9) und die Produktgruppe soll eine Klassifizierung nach Qualität oder Größe ermöglichen.

Da Erzeugnisse auf mehreren Betriebsmitteln und in verschiedenen Funktionsbereichen gefertigt werden können, müssen sie in einem Transporthilfsmittel⁶ (THM) zum nächsten Arbeitsplatz transportiert werden. Deshalb ist es nötig, die Füllstückzahl⁷ und den Namen des Transporthilfsmittels für jedes Erzeugnis zu speichern. Die Erstellungsart gibt an, ob es in der Fabrik produziert oder fremdbezogen wird. Über die Bezeichnung kann das Erzeugnis genauer beschrieben werden. Erzeugniselemente werden ebenfalls über das Erzeugnis angegeben und unter Typ als Baugruppe, Einzelteil usw. vermerkt.

Im Prototyp können einem Erzeugniselement mehrere Prozesse zugeordnet werden, jedoch geht dabei nicht ihre Reihenfolge hervor. Ermöglicht werden kann das durch die Einführung der Entity **Arbeitsplan** (Tabelle 10). Ein Arbeitsplan beschreibt die „Vorgangfolgen zur Fertigung einer Komponente oder eines Erzeugnisses“ [Zäp01, Seite 71 f.]. Es werden die Reihenfolge der Arbeitsvorgänge, die dazugehörige Materialnummer,

⁶ „Transporthilfsmittel sind alphanumerische Elemente, die eingesetzt werden, wenn mehrere Artikel eines Artikeltyps zusammen von einer Station zur nächsten transportiert werden sollen.“ [All99, Seite 98]

⁷ Die Anzahl der Erzeugniselemente, die ein THM füllen.

das verwendete Betriebsmittel und die Technologie angegeben. Außerdem werden Rüst- und Stückzeiten⁸ sowie Losgrößen erfasst. Die Entity Prozess wird nicht mehr benötigt, da alle erforderlichen Informationen zu dem Prozess von dem Arbeitsplan erfasst werden. Der Name wird nicht übernommen, da dem Arbeitsplan keiner zugewiesen werden muss. Er identifiziert sich über die Materialnummer und den Arbeitsvorgang. Die Bearbeitungszeit und –intensität werden durch die Rüst- und Stückzeit abgelöst. Informationen zu Energie- und CO₂-Verbrauch sind nicht Bestandteil des Arbeitsplans und werden deshalb nicht übernommen.

Tabelle 9: Entity Erzeugnis

Prototyp	Erweiterung		
Erzeugnis	Erzeugnis		
Attribut	Attribut	Datentyp	Bemerkung
<u>eid</u>			
Name	<u>Materialnummer</u> (12)	char	Einzelfertigung, Klein-/Mittel-/Großserie, Massenfertigung
Fertigungsart	Fertigungsart (20)	char	Stunden [0;24]
Tagesarbeitszeit	Tagesarbeitszeit	float	Stück pro Jahr
Produktionsmenge	Produktionsmenge	int	
	Produktgruppe (20)	char	Klassifizierung nach Qualität, Größe usw.
	Bezeichnung	varchar	
	Erstellungsart (20)	char	Eigenfertigung/Fremdfertigung
	Typ (20)	char	Baugruppe, Einzelteil
	Transporthilfsmittelname (20)	char	
	Füllstückzahl	int	Stück
	Häufigkeit	int	Prozent
	Transportlosgröße	int	Stück

Bei der Planung von Fabriken werden u.a. Tabellen, Zeichnungen und Grafiken erstellt, welche in der Datenbank gespeichert werden sollen, um jederzeit darauf zugreifen zu können. Im Prototyp wurde dies noch nicht berücksichtigt, in der Erweiterung soll jedoch die Möglichkeit bestehen, Dokumente zu speichern (Tabelle 11), damit sie jederzeit zur Verfügung stehen. Das können z.B. Anleitungen zu einem Betriebsmittel,

⁸ „Die Losgröße ist die Mindestmenge, in der ein Teil beschafft oder gefertigt wird.“ [GK07, Seite 683]

Zeichnungen zu Produkten, Beschreibungen zum Arbeitsplan und Grundrisse von Gebäuden sein.

Um ein Dokument in einer Datenbank zu speichern, gibt es verschiedene Möglichkeiten. Indem ein Dokumentenverwaltungssystem verwendet wird, sind die Suche und das Wiederfinden von Dokumenten einfach. Eine andere Möglichkeit ist die Speicherung als Binary Large Object (BLOB). Dieser Datentyp wurde entwickelt, um große Objekte wie Bilder, Audio- und Videosequenzen [SHS05, Seite 106] zu speichern. Dadurch kann jedoch die Datenbank schnell sehr groß werden. Ein Vorteil ist die konsistente Datensicherung. Eine weitere Möglichkeit ist, den Pfad des Dokumentes in einem Textfeld zu speichern. Das hat den Nachteil, dass die Suche nach einem Dokument erschwert wird. Da die Entwicklung des Werkzeugs noch am Anfang steht, ist es ausreichend den Name und Pfad, wo das Dokument gespeichert werden soll, zu hinterlegen. In Weiterentwicklungen, wenn der Umfang der Anwendung zunimmt, kann die Einführung eines Dokumentenverwaltungssystems in Betracht gezogen werden, da es die genannten Vorteile mitbringt.

Tabelle 10: Entity Arbeitsplan

Prototyp	Erweiterung		
Prozess	Arbeitsplan		
Attribut	Attribut	Datentyp	Bemerkung
<u>pid</u>	<u>Materialnummer</u> (12)	char	
	<u>Arbeitsvorgang</u>	int	
	Betriebsmittel	char	
	Technologie	char	
	Rüstzeit	int	Sekunden
	Stückzeit	int	Sekunden
	Losgröße	int	Stück
Name			
Bearbeitungszeit			
Bearbeitungsintensität			
Energie			
CO2			

Tabelle 11: Entity Dokument

Prototyp	Erweiterung		
-	Dokument		
Attribut	Attribut	Datentyp	Bemerkung
	<u>Bezeichnung</u> (20)	char	
	Objekt	char	Auswahl nach der DIN 8580
	Beschreibung	text	
	Pfad	text	

Nachdem die Grunddaten betrachtet wurden, werden im Folgenden die entstehenden Ergebnisdaten beschrieben. Wird eine Fabrik neu geplant oder umstrukturiert, werden die Anzahl der benötigten Betriebsmittel (**Berechnungsergebnisse**), die daraus entstehenden Investitionskosten und **Szenarien** berechnet. Mit Hilfe des Werkzeugs soll außerdem der Materialfluss (**Transportintensitäten**) zwischen Betriebsmitteln und Funktionsbereichen berechnet werden.

Für jeden Funktionsbereich wird die benötigte Anzahl der Betriebsmittel errechnet und die Auslastung je Betriebsmitteltyp in einem Funktionsbereich sowie die entstehenden Investitionskosten und Betriebskosten bestimmt (Tabelle 12). Im Prototyp gibt es keine Entity, die die Anzahl der Betriebsmittel je Funktionsbereich speichert.

Tabelle 12: Entity Berechnungsergebnisse

Prototyp	Erweiterung		
-	Berechnungsergebnisse		
Attribut	Attribut	Datentyp	Bemerkung
	<u>Funktionsbereich</u> (12)	char	
	<u>Betriebsmittel</u> (20)	char	
	Anzahl	int	Stück
	Auslastung	float	
	Investitionskosten	float	Euro
	Betriebskosten	float	Euro

Die möglichen Anordnungen der Funktionsbereiche in den Betrieben werden als **Szenarien** bezeichnet. Sie stellen die verschiedenen Lösungsvarianten aus der Phase der Konzeptplanung im Fabrikplanungsprozess dar. Im Prototyp ist eine Entity Szenario

bereits vorhanden. Jedoch speichert sie Informationen zu Baustrukturen. Tabelle 13 zeigt, dass nur der Name übernommen wird. Zu jedem Funktionsbereich wird seine Fläche hinterlegt. Damit er später angeordnet werden kann, ist die Form der Fläche wichtig. Deshalb werden auch die Länge und Breite des Funktionsbereiches erfasst. Die Anordnung erfolgt auf einer zweidimensionalen Fläche, die durch X- und Y-Koordinaten gekennzeichnet ist. Sie dienen als Information wo der Bereich angeordnet wurde.

Da die Szenarien visualisiert werden, können alle Informationen bzgl. des Szenarios in einem XML-Dokument gespeichert werden. In dem XML-Dokument sollen mindestens die Abmessungen und Positionen der Funktionsbereiche in den Betrieben gespeichert werden.

Tabelle 13: Entity Szenario

Prototyp	Erweiterung		
Szenario	Szenario		
Attribut	Attribut	Datentyp	Bemerkung
<u>sid</u>	<u>ID</u>	int	
	Funktionsbereich (20)	char	
Name	Bezeichnung (20)	char	
	Betrieb (20)	char	
	XML	clob	
	Fläche	float	Meter ²
	X-Koordinate	float	
	Y-Koordinate	float	
	Länge	float	Meter
	Breite	float	Meter
	Betrieb		
	Transportaufwand	float	
bsid			

Die vorliegende relationale Datenbank weist eine flache Tabellenstruktur auf. XML-Dokumente weisen dagegen eine hierarchische Schachtelung auf und sind beliebig tief strukturiert [Kud07, Seite 373; End0, Seite 360]. Es werden Elemente sowie Attribute unterschieden und Kommentare treten an verschiedenen Stellen auf [Kud07, Seite 373].

Zur Speicherung von XML-Dokumenten in einer relationalen Datenbank gibt es nach [Kud07, Seite 372] drei Ansätze.

Erstens können XML-Dokumente inhaltsorientiert in ein Netz von Tabellen zerlegt werden. Die Methode wird verwendet, um eine relationale Tabelle als XML-Dokument darzustellen, welche eine regelmäßige Struktur und sehr geringe Tiefe hat. Bei diesem Ansatz werden nur der Inhalt und nicht die Auszeichnungselemente (markup) abgebildet. Der Vorteil dieser Möglichkeit ist, dass die Daten von dem DBMS und SQL verwendet werden können. Ein Element im XML-Dokument wird zu einer Spalte in einer Tabelle. Jedoch gehen dabei die Reihenfolge der Elemente sowie Kommentare und Verarbeitungsanweisungen verloren. Auch werden komplexere XML-Schema-Konstrukte nicht unterstützt. Die zweite Möglichkeit ist es, durch strukturorientierte Zerlegung das Dokument verlustfrei abzubilden, indem auch die Struktur vollständig hinterlegt wird. Jeder Knoten und jede Kante wird auf eine Tabellenzeile abgebildet. Das ermöglicht die vollständige Abbildung des XML-Dokuments. Der Nachteil ist, dass dabei viele Tabellenzeilen entstehen und der Aufwand bei großen XML-Dokumenten steigt. Außerdem ist ein gewisser Informationsverlust möglich. Bei der dritten Möglichkeit wird das gesamte XML-Dokument in einer einzigen Spalte vom Typ `varchar` oder Character Large Object (CLOB) gespeichert. Ein CLOB, wurde, ähnlich einem BLOB, entwickelt, um große Objekte vom Typ `char` zu speichern [SHS05, Seite 106; Kud07, Seite 374 f.]. Dieses lässt sich einfach realisieren und das DBMS kann mit Hilfe von Zeigern oder des BLOB-Directorys auf Teile des Dokumentes zugreifen [SHS05, Seite 106 ff.].

Die XML-Dokumente, die in dieser Datenbank hinterlegt werden sollen, dienen der Visualisierung der in ihnen gespeicherten Daten. Die Speicherung als CLOB bietet die Möglichkeit den Aufbau des Dokumentes zu sichern, wodurch im Vergleich zu den ersten beiden Methoden ein gewisser Informationsverlust vermieden werden kann [En01, Seite 361].

Bei Dokumenten, die der Visualisierung dienen, spielt die Einhaltung der Elementreihenfolge eine große Rolle.

Dadurch, dass Erzeugnisse auf unterschiedlichen Betriebsmitteln in verschiedenen Bereichen gefertigt werden, entstehen Transportintensitäten (Materialfluss). In der Datenbank sollen Transportintensitäten bzgl. Betriebsmittel und Funktionsbereichen hinter-

legt werden. Dazu werden zu jedem Funktionsbereich und Betriebsmittel der Quell- und Zielbereich sowie die Menge der zu transportierenden Einheiten gespeichert.

Nachdem die zu speichernden Grunddaten und Ergebnisdaten beschrieben wurden, die in der Datenbank gespeichert werden sollen, werden im Folgenden die Beziehungen zwischen ihnen im Vordergrund stehen.

Folgende Beziehungen gelten zwischen den Grunddaten:

- Ein Werk **besteht aus** N Betrieben. Ein Betrieb ist genau einem Werk zugeordnet (1:N-Beziehung).
- Ein Betrieb ist in N Funktionsbereiche **gegliedert**. Ein Funktionsbereich ist genau einem Betrieb zugeordnet (1:N-Beziehung).
- Ein Funktionsbereich **beinhaltet** N Betriebsmittel. Ein Betriebsmittel wird M Funktionsbereichen zugeordnet (M:N-Beziehung). Die Beziehung nimmt das Attribut Anzahl auf.
- Eine Technologie wird von N Betriebsmitteln **verwendet**. Ein Betriebsmittel kann M Technologien verwenden (M:N-Beziehung).
- Jeder Eintrag im Arbeitsplan **enthält** genau ein Betriebsmittel. Ein Betriebsmittel ist in N Arbeitsplänen aufgeführt (1:N-Beziehung).
- Ein Arbeitsplan **gilt für** genau ein Erzeugnis. Ein Erzeugnis ist in einem Arbeitsplan bei mehreren Arbeitsvorgängen aufgeführt (1:N-Beziehung).
- Ein Werk **stellt** N Erzeugnisse **her**. Ein Erzeugnis wird genau einem Werk zugeordnet (1:N-Beziehung).
- Zu einem Werk werden N Dokumente **gespeichert**. Jedes Dokument wird genau einem Werk zugeordnet (1:N-Beziehung).
- Ein Betrieb **hat** mehrere Etagen. Jede Etage wird einem Betrieb zugeordnet (1:N-Beziehung).

Eine weitere Beziehung zwischen den Grunddaten gilt zwischen Funktionsbereich und Erzeugnis. Im Prototyp wurden alle Strukturierungsprinzipien außer der Produktorientierung (siehe Abbildung 3-4) realisiert. Um die Segmentierung aufgrund der zu fertigenden Erzeugnisse zu ermöglichen, werden die Erzeugnisse nicht nur dem Werk, sondern auch dem Funktionsbereich zugeordnet [Sch10, Seite 24 f.]:

- Ein Erzeugnis wird N Funktionsbereichen zugeordnet. Ein Funktionsbereich **produziert** M Erzeugnisse (M:N-Beziehung).

Damit es möglich ist, Stücklisten für die Erzeugnisse abzubilden, wird eine rekursive Beziehung für das Erzeugnis eingeführt. Damit lässt sich die Hierarchie nachbilden, die das Merkmal einer Stückliste ist.

- Ein Erzeugnis **ist Teil von** N Erzeugnissen bzw. ihnen untergeordnet. N Erzeugnisse sind wiederum M Erzeugnissen übergeordnet. Zu der Beziehung werden außerdem die Attribute Anzahl und Häufigkeit angegeben.

Um Ergebnisse bzgl. Planungen zu speichern, werden die folgenden Beziehungen benötigt:

- Einem Funktionsbereich werden mehrere Berechnungsergebnisse **zugeordnet**. Jedem Berechnungsergebnis wird ein Funktionsbereich zugeordnet (1:N-Beziehung).
- Dem Funktionsbereich sind N Szenarien **zugeordnet**. Ein Szenario ist genau einem Funktionsbereich zugeordnet (1:N-Beziehung) und wird in der VDT-Plattform visualisiert.

Die Transportintensitäten (Materialfluss) bzgl. der Funktionsbereiche und Betriebsmittel werden jeweils über rekursive Beziehungen abgebildet.

- Ein Funktionsbereich **liefert an** N Funktionsbereiche Erzeugnisse. Die Funktionsbereiche liefern wiederum an M Funktionsbereiche Erzeugnisse weiter (rekursive M:N-Beziehung).
- Ein Betriebsmittel **liefert an** N Betriebsmittel Erzeugnisse. Die Betriebsmittel liefern wiederum an M Betriebsmittel Erzeugnisse weiter (rekursive M:N-Beziehung).

Es kann ein relationales Datenbanksystem und ein Microsoft-SQL-Server 2005⁹ als DBMS verwendet werden. Durch die Verwendung eines relationalen DBMS und deren Eigenschaften bzgl. Datenunabhängigkeit, besteht jederzeit die Möglichkeit das Schema für die Datenbank zu erweitern, d.h. neue Tabellen können nach Bedarf hinzugefügt, gelöscht oder verändert werden.

Daten zu Personal-, Informations- sowie Ver- und Entsorgungsflüssen werden zunächst nicht betrachtet, da diese in den konzeptionellen Arbeiten nicht weiter berücksichtigt wurden. Des Weiteren werden Aspekte der Standortplanung ausgenommen und nur die Fälle der Neu- und Umplanung einbezogen.

⁹ <http://www.microsoft.com/sqlserver/2005/en/us/product-information.aspx>

4.1.2 Verarbeitungsanforderungen

Durch Integritätsanalysen soll sichergestellt werden, dass für entsprechende Funktionen die benötigten Daten vorliegen. Eine weitere Anforderung an die Datenbank ist ihre Erweiter- und Änderbarkeit. Auf der Datenbank müssen Änderungen am Datenbestand wie Einfügen, Aktualisieren und Löschen der Daten vorgenommen werden können, damit der Planer im Programm einen Betrieb hinzufügen, bearbeiten, löschen oder aktualisieren kann. Das gilt auch für die anderen Entities. So sollen z.B. neue Dokumente gespeichert oder gelöscht, neue Szenarien generiert und Arbeitspläne angelegt werden können.

Wenn erforderliche Eingaben vorzunehmen sind, bietet sich die Spezifikation von Standardwerten an, die im Bedarfsfall angepasst werden müssen. Es soll die Möglichkeit bestehen, jederzeit die Grunddaten zu ändern.

Wie in Kapitel 4.1.1 beschrieben, soll durch die Angabe einer Technologie eine Vorauswahl bzgl. des Betriebsmittels getroffen werden. Dazu muss auf der Datenbank eine automatisierte Anfrage ausgeführt werden, die anhand der Technologie die Auswahl der Betriebsmittel einschränkt.

Die Tagesarbeitszeit und die Fertigungsart sollen für die Entities Erzeugnis, Funktionsbereich und Betriebsmittel vom Betrieb übernommen werden, aber jederzeit anpassbar sein. Werden nun die Tagesarbeitszeit und Fertigungsart auf der Ebene des Betriebs geändert, sollen die angepassten Werte im Erzeugnis, Funktionsbereich und Betriebsmittel unverändert bleiben.

Eine weitere durchzuführende Abfrage, ist die Überprüfung der Quell- und Zielfunktionsbereiche bzw. -betriebsmittel in den Entities der Transportintensitäten für Funktionsbereiche bzw. Betriebsmittel. Es soll nicht möglich sein, in beide Spalten dieselben Namen von Funktionsbereichen bzw. Betriebsmitteln einzutragen, da ein Bereich bzw. eine Maschine nicht mit sich selbst verkettet ist.

Zudem sollen aus der Produktionsmenge der Erzeugnisse sowie der hinterlegten Stücklisten die benötigten Mengen der Baugruppen und Einzelteile berechnet werden.

4.2 Konzeptioneller Entwurf

Aus den Anforderungen aus Abschnitt 4.1 wird im Folgenden ein zielsystemunabhängiges Datenbankschema erstellt. Darin werden die Entities und ihre Beziehungen abgebildet. Abbildung 4-1 zeigt einen Entwurf für das konzeptionelle Schema.

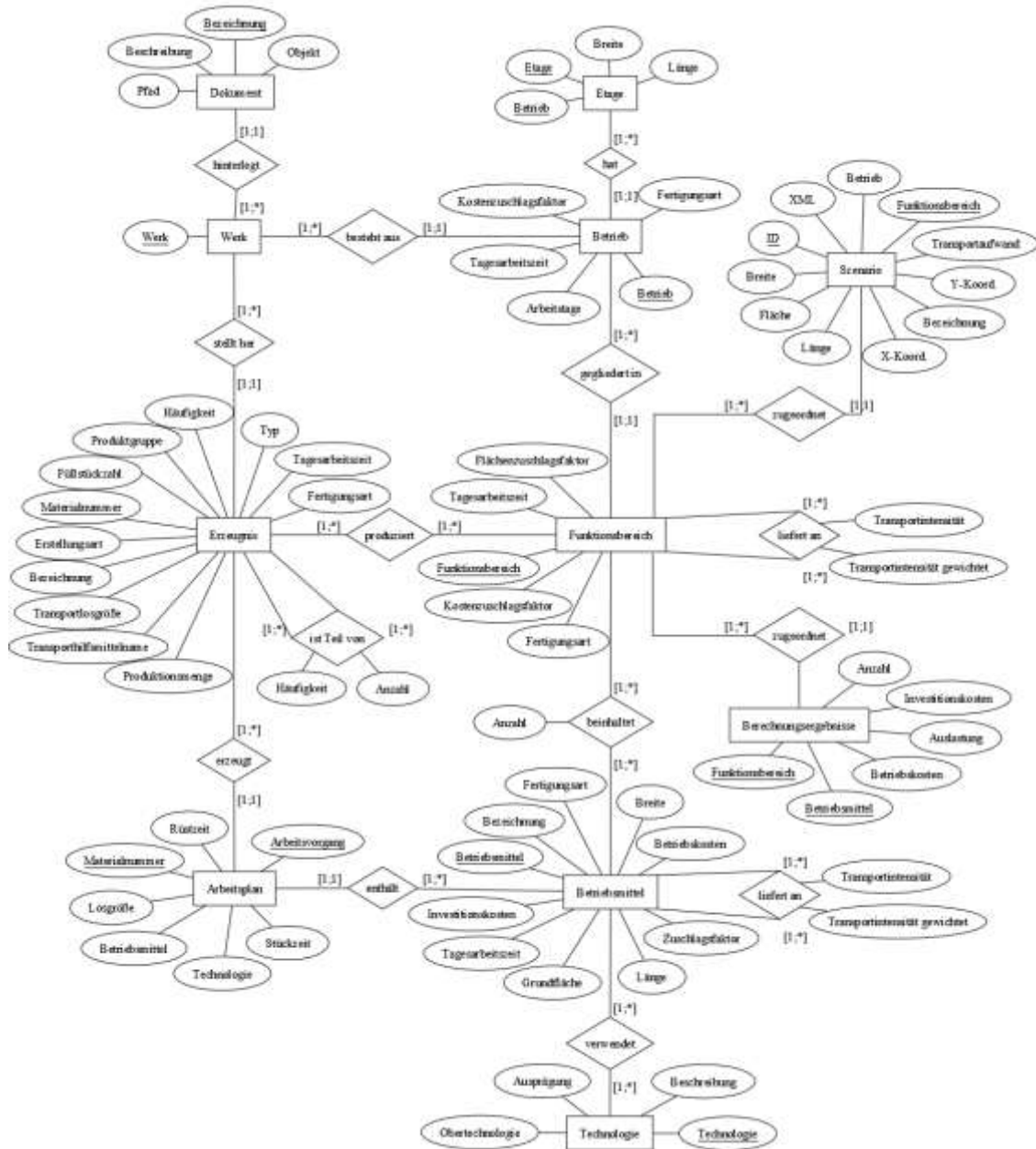


Abbildung 4-1: Konzeptioneller Entwurf

Ein Werk besteht aus mehreren Betrieben, die wiederum in Funktionsbereiche gegliedert sind. Ein Werk produziert außerdem Erzeugnisse, die sich wieder aus Erzeugnissen zusammensetzen. Um die Hierarchie einer Stückliste darzustellen, wurde eine rekursive Beziehung für das Erzeugnis eingeführt. Das bedeutet, dass jede Baugruppe und Einzelteil als Erzeugnis angegeben wird. Mit Hilfe der rekursiven Beziehung wird jeweils das

übergeordnete Element zugeordnet. Diese können in mehreren Funktionsbereichen und auf unterschiedliche Weisen gefertigt werden. Der Arbeitsplan beinhaltet die Arbeitsvorgänge sowie die zu verwendenden Betriebsmittel. Diese werden einem Funktionsbereich und einer Technologie zugeordnet. Zu einem Werk werden außerdem Dokumente gespeichert.

Die Transportintensitäten der Funktionsbereiche und Betriebsmittel werden ebenfalls als rekursive Beziehungen dargestellt, um den Materialfluss abzubilden. Für jeden Funktionsbereich werden die benötigte Betriebsmittellanzahl und deren Auslastung sowie Investitionskosten berechnet. Im Szenario werden die Funktionsbereiche mit ihren verschiedenen Möglichkeiten der Anordnung hinterlegt.

Das ER-Modell des Prototyps wurde um die Entities Werk, Dokument, Szenario, Berechnungsergebnisse und Arbeitsplan sowie um die zugehörigen Beziehungen erweitert. Im Prototyp wurden Erzeugnisse und Erzeugniselemente voneinander getrennt. In der Erweiterung dagegen werden Informationen zu beiden in einer Tabelle abgespeichert und in einer rekursiven Beziehung die Stückliste abgebildet.

4.3 Logischer Entwurf

Im logischen Entwurf wird aus dem konzeptuellen Schema aus Kapitel 4.2 ein relationales Schema entwickelt [Kud07, Seite 81]. Da das ER-Modell aus Entities und Beziehungen besteht und dem im relationalen Modell nur die Relation gegenübersteht, werden beide Typen im ersten Schritt auf jeweils eine Relation abgebildet. Dafür sind Regeln erforderlich, die die Abbildung der Einheiten der Datenmodelle definieren [KE01, Seite 71]. Die Regeln werden im Folgenden vorgestellt.

Jede Entity wird auf jeweils ein Relationenschema abgebildet und dessen Attribute und Schlüssel übernommen [SSH08, Seite 148]. So wird z.B. aus der Entity Betrieb aus Abbildung 4-1 die folgende Relation Betrieb gebildet:

$$\mathbf{Betrieb} = \{\underline{\mathbf{Betrieb}}, \text{Arbeitstage, Fertigungsart, Tagesarbeitszeit, Kostenzuschlagsfaktor}\}$$

Die Vorgehensweise für die anderen Entities erfolgt analog.

Beziehungen werden ebenfalls auf jeweils ein Relationenschema abgebildet. Die Attribute der Beziehung und die Primärschlüssel der beteiligten Entities werden zugeordnet

[SSH08, Seite 149]. Der Relationenname entspricht dabei dem Namen der Beziehung. Fremdschlüsselbedingungen werden durch einen Pfeil dargestellt.

So wird bspw. aus der Beziehung „stellt her“ die folgende Relation „stellt_her“ erstellt:

$$\text{stellt_her} = \{\text{Werk} \rightarrow \text{Werk}, \text{Materialnummer} \rightarrow \text{Erzeugnis}\}$$

Die Wahl des Schlüssels bei der Abbildung der Beziehung hängt von der Kardinalität der Beziehung ab. Die Schlüssel werden wie folgt festgelegt:

- M:N-Beziehung: beide Primärschlüssel zusammen werden Schlüssel im neuen Relationenschema.
- 1:N-Beziehung: Primärschlüssel der N-Seite wird Schlüssel im neuen Relationenschema.
- 1:1-Beziehung: beide Primärschlüssel werden je ein Schlüssel im neuen Relationenschema, der Primärschlüssel wird dann aus diesen Schlüsseln gewählt [SSH08, Seite 149].

Demnach wird in der Relation „produziert“ die N-Seite (Erzeugnis) Schlüssel:

$$\text{stellt_her} = \{\text{Werk} \rightarrow \text{Werk}, \underline{\text{Materialnummer}} \rightarrow \text{Erzeugnis}\}$$

Die Transformation der anderen Beziehungen erfolgt analog.

Eine Besonderheit bei der Abbildung von Beziehungen stellt die rekursive Beziehung dar. Dabei werden die beteiligte Entity und die Beziehung auf jeweils ein Relationenschema abgebildet. Die Beziehung besitzt zwei Primärschlüssel, die die Namen der Rollen erhalten. Es ist nicht möglich den Schlüssel der Entity zu übernehmen, da die Relation der Beziehung nicht zwei Primärschlüssel mit denselben Namen haben darf. Die rekursive Beziehung „ist Teil von“ wird auf eine neue Relation abgebildet und enthält die Primärschlüssel Überelement und Unterelement:

$$\text{ist_Teil_von} = \{\underline{\text{Überelement}} \rightarrow \text{Erzeugnis}, \underline{\text{Unterelement}} \rightarrow \text{Erzeugnis}, \text{Anzahl}, \text{Häufigkeit}\}$$

Die Attribute Überelement und Unterelement sind Fremdschlüssel auf das Attribut Materialnummer in der Relation Erzeugnis.

In einigen Fällen können Relationenschemata von Entities und Beziehungen verschmolzen werden. Zwischen den verbleibenden Relationenschemata werden Fremdschlüsselbedingungen eingeführt. Bei optionalen Beziehungen [0,1] oder [0,N] werden keine Verschmelzungen durchgeführt. Bei zwingenden Beziehungen wie z.B. [1,1] oder [1,N] können die Relationen verschmolzen werden [SSH08, Seite 149].

- 1:N-Beziehung: die N-Seite wird mit der Relation der Beziehung verschmolzen.
- 1:1-Beziehung: beide Entities werden in das Relationenschema der Beziehung integriert [SSH08, Seite 149 f.].

Das gesamte Relationenschema ist im Folgenden aufgelistet:

Werk = {Werk}

Dokument = {Bezeichnung, Objekt, Beschreibung, Pfad, Werk → Werk}

Betrieb = {Betrieb, Arbeitstage, Fertigungsart, Tagesarbeitszeit, Kostenzuschlagsfaktor, Werk → Werk}

Funktionsbereich = {Funktionsbereich, Fertigungsart, Tagesarbeitszeit, Flächenzuschlagsfaktor, Kostenzuschlagsfaktor, Betrieb → Betrieb}

beinhaltet = {Funktionsbereich → Funktionsbereich, Betriebsmittel → Betriebsmittel, Anzahl}

Betriebsmittel = {Betriebsmittel, Tagesarbeitszeit, Fertigungsart, Bezeichnung, Betriebskosten, Grundfläche, Zuschlagsfaktor, Länge, Breite, Investitionskosten}

Etage = {Betrieb → Betrieb, Etage, Länge, Breite}

verwendet = {Betriebsmittel → Betriebsmittel, Technologie → Technologie}

Technologie = {Technologie, Beschreibung, Ausprägung, Obertechnologie}

Bei der Abbildung der rekursiven Beziehungen „liefert an“ werden die Primärschlüssel Funktionsbereich bzw. Betriebsmittel jeweils in „Quelle“ und „Ziel“ umbenannt. Das verdeutlicht zusätzlich den Materialfluss.

liefert_an_Betriebsmittel = {Quelle → Betriebsmittel, Ziel → Betriebsmittel, Transportintensität, Transportintensität_gewichtet}

liefert_an_Funktionsbereich = {Quelle → Funktionsbereich, Ziel → Funktionsbereich, Transportintensität, Transportintensität_gewichtet}

Erzeugnis = {Materialnummer, Tagesarbeitszeit, Fertigungsart, Produktionsmenge, Produktgruppe, Bezeichnung, Erstellungsart, Typ, Transporthilfsmittelname, Füllstückzahl, Häufigkeit, Transportlosgröße}

istTeilvon = {Überelement → Erzeugnis, Unterelement → Erzeugnis, Häufigkeit, Anzahl}

produziert = {Materialnummer → Erzeugnis, Funktionsbereich → Funktionsbereich}

Arbeitsplan = {Materialnummer, Arbeitsvorgang, Betriebsmittel → Betriebsmittel, Technologie → Technologie, Rüstzeit, Stückzeit, Losgröße}

Szenario = {ID, Funktionsbereich → Funktionsbereich, Betrieb → Betrieb, Fläche, X-Koordinate, Y-Koordinate, Länge, Breite, Transportaufwand, XML, Bezeichnung}

Berechnungsergebnisse = {Funktionsbereich → Funktionsbereich, Betriebsmittel → Betriebsmittel, Anzahl, Auslastung, Investitionskosten, Betriebskosten}

In der Relation Szenario sind die Attribute Fläche, Länge und Breite nur von dem Funktionsbereich abhängig. Um Redundanzen zu vermeiden, werden diese Attribute in die Relation Funktionsbereich aufgenommen. Die geänderten Relationen Szenario und Funktionsbereich sind im Folgenden aufgelistet:

Szenario = {ID, Name → Funktionsbereich, Betrieb, X-Koordinate, Y-Koordinate, Transportaufwand, XML, Bezeichnung}

Funktionsbereich = {Funktionsbereich → Funktionsbereich, Fertigungsart, Tagesarbeitszeit, Flächenzuschlagsfaktor, Kostenzuschlagsfaktor, Betrieb → Betrieb, Fläche, Länge, Breite}

5 Evaluierung

In diesem Kapitel wird das konzeptuelle Schema aus Kapitel 4.2 evaluiert. Zunächst soll anhand des Relationenmodells aus Kapitel 4.3 überprüft werden, inwiefern die Anforderungen aus Kapitel 4.1 umgesetzt wurden. Um das Relationenschema zu analysieren, wird die Normalisierung der Daten in Kapitel 5.2 durchgeführt.

5.1 Umsetzung der Anforderungen

Die Anforderungen aus Kapitel 4.1 werden untersucht, um anschließend auswerten zu können, inwiefern diese im konzeptionellen Entwurf umgesetzt wurden.

- Die Erweiterung erfolgt auf Werksebene.

In der Anforderungsanalyse wurde zunächst gefordert, dass die Erweiterung nicht mehr einen Betrieb betrachtet, sondern ein Werk. Deshalb wurde die Entity Werk, die aus mehreren Betrieben bestehen kann, eingeführt.

- Die Entities werden aus dem Prototyp teilweise übernommen und der Erweiterung des Prototyps angepasst.

Die Entities Betrieb, Funktionsbereich, Erzeugnis, Technologie, Betriebsmittel und Szenario wurden aus dem Prototyp mit den entsprechenden Änderungen übernommen. So wurde in der Entity Betrieb zusätzlich der Kostenzuschlagsfaktor eingeführt, um Kosten für zusätzliche Flächen zu berücksichtigen. Der Funktionsbereich wurde um die Attribute Flächen- und Kostenzuschlagsfaktor erweitert, um zusätzliche Flächen für Lager und Transport sowie zusätzliche Kosten zu berücksichtigen. Durch die Attribute Transporthilfsmittelname und Füllstückzahl in der Entity Erzeugnis werden zusätzliche Informationen für den Transport von Erzeugniselementen zwischen den Betriebsmitteln bereitgestellt. Die Tagesarbeitszeit und die Fertigungsart werden entsprechend der konzeptionellen Erweiterung auch dem Betriebsmittel zugeordnet.

- Der konzeptionellen Erweiterung folgend, werden die Entities Dokument, Arbeitsplan, Berechnungsergebnisse und Etage eingeführt.

Die Entity Arbeitsplan ersetzt den Prozess, welcher in der Erweiterung als Eigenschaft eines Arbeitsplanes in der Tabelle auftritt. Die Entity Arbeitsplan besitzt im Gegensatz zum Prozess zusätzlich die Attribute Rüstzeit, Stückzeit und Losgröße. Durch die Erweiterung wurde gefordert, dass Dokumente, Szenarien, Flächen und Arbeitspläne in der Datenbank gespeichert werden können. Indem jeweils neue Entities diesbezüglich

eingeführt wurden, erfüllt das Modell die Anforderungen. Somit können Zeichnungen, Tabellen und Grafiken hinterlegt sowie Szenarien abgelegt und bewertet werden. Die Entity Etage (Fläche) ermöglicht es die Größe der Betriebe anzugeben, um dann später die Funktionsbereiche anzuordnen. Durch die Entity Arbeitsplan ist es möglich, mehrstufige Fertigungen zu planen.

- Die Beziehungen zwischen den Entities werden an die Erweiterung angepasst.

Für die neuen Entities Dokument, Etage, Szenario und Berechnungsergebnisse wurden die entsprechenden Beziehungen eingeführt. Durch die Ersetzung des Prozesses durch den Arbeitsplan bleibt die Beziehung zu dem Betriebsmittel, jedoch wird in der Erweiterung die Verbindung zu dem Erzeugnis hergestellt. Indem das Erzeugnis dem Funktionsbereich zugeordnet wurde, kann eine produktorientierte Planung erfolgen. Des Weiteren wurden rekursive Beziehungen bzgl. Erzeugnis, Funktionsbereich und Betriebsmittel eingeführt, um die Hierarchie darzustellen.

- Es soll die Integrität der Daten gesichert werden.

Die referentielle Integrität wird durch Fremdschlüssel gesichert. Beim Löschen bzw. Aktualisieren werden alle verknüpften Datensätze automatisch gelöscht bzw. aktualisiert. Außerdem kann der Fremdschlüssel nur solche Werte annehmen, die der Primärschlüssel besitzt. Dadurch werden Eingaben, die nicht möglich sind, vermieden. Weiterhin wird die Integrität durch Festlegen des Wertebereichs gesichert. So stehen bei der Eingabe der Fertigungsart die folgenden Möglichkeiten zur Auswahl: Einzelfertigung, Klein-/ Mittel-/ Großserie oder Massenfertigung.

5.2 Normalform

In diesem Kapitel wird untersucht, inwiefern das Relationenmodell aus Kapitel 4.3 den Normalformen einer relationalen Datenbank entspricht. Der Vorgang nennt sich Normalisierung und dabei werden durch Aufspalten von Relationenschemata Redundanzen vermieden, ohne gleichzeitig semantische Informationen zu verlieren (Abhängigkeits-treue) und die Möglichkeit zur Rekonstruktion der Relationen zu verlieren (Verbund-treue) [SSH08, Seite 183 ff.].

Normalformen werden angewendet, um Eigenschaften wie minimale Redundanz sowie Minimierung der Einfügungs-, Lösch- und Update-Anomalien zu erreichen. Relationenschemata, die die Bedingungen der Normalformen nicht erfüllen, werden in

kleinere Relationenschemata zerlegt [SSH08, Seite 182]. Es muss nicht bis zur höchstmöglichen Normalform normalisiert werden. Aus Performanzgründen können Relationen in einer niedrigeren Normalisierungsform bleiben [EN02, S.519].

Das Ziel der Normalisierung ist der verlustfreie Join, um damit zu gewährleisten, dass in den nach der Zerlegung erstellten Relationenschemata keine unechten Tupel erzeugt werden. Die Abhängigkeitswahrung gewährleistet, dass jede funktionale Abhängigkeit nach der Zerlegung in mindestens einer Relation dargestellt wird [EN02, Seite 518].

Im Folgenden werden die Normalformen kurz vorgestellt und das Relationenschema daraufhin analysiert, ob es die jeweilige Normalform erfüllt.

1. Normalform

In der ersten Normalform sind keine nicht-atomaren Attribute oder verschachtelte Relationen enthalten. Z.B. sollte Adresse nicht als Attribut verwendet werden, da es in Straße, PLZ, Ort aufgespalten werden kann [SSH08, Seite175 f.].

Im Relationenschema aus Kapitel 4 sind keine nicht-atomaren Attribute vorhanden, können also nicht mehr aufgespalten werden. Es liegt demnach in der ersten Normalform vor.

2. Normalform

Eine Relation ist in der zweiten Normalform, wenn sie in der ersten Normalform vorliegt und jedes Nichtschlüsselattribut von allen Schlüsseln und nicht nur von einem Teil abhängig ist [SSH08, Seite 176 f.].

3. Normalform

In der dritten Normalform sind keine transitiven Abhängigkeiten erlaubt. Diese liegt vor, wenn ein Schlüssel eine Attributmenge und diese wiederum eine andere Attributmenge des gleichen Schemas funktional bestimmt [SSH08, Seite179].

Das Relationenschema aus Kapitel 4 liegt in der dritten Normalform vor. Die Attribute sind atomar, können also nicht mehr aufgespalten werden und sind jeweils von allen Schlüsseln der jeweiligen Relation abhängig. Des Weiteren liegen keine transitiven Abhängigkeiten vor.

6 Zusammenfassung und Ausblick

In dieser Arbeit wurden zunächst die Grundlagen von Datenbanksystemen behandelt. Das vorgestellte Phasenmodell für den Datenbankentwurf dient als Vorgehensweise für das Datenbank-Konzept in Kapitel 4. Mit Hilfe des Prototyps wurde die Phase der Konzeptplanung im Fabrikplanungsprozess erläutert und die Funktionsweise des Werkzeugs erklärt.

Die konzeptionelle Arbeit zur Erweiterung von Quovadis wurde analysiert und daraus die Anforderungen an die erweiterte Datenbank in Kapitel 4.1 abgeleitet. Dabei wurde der Prototyp der Erweiterung gegenübergestellt und verglichen. Es wurde ersichtlich, welche Informationen aus dem Prototyp übernommen werden konnten und welche hinzugekommen sind. Im konzeptionellen Entwurf wurden die Informationsanforderungen in einem Schema visualisiert, um dann im logischen Entwurf in ein relationales Schema transformiert zu werden.

Mit Hilfe der Evaluierung wurde gezeigt, dass die Anforderungen in das Schema umgesetzt wurden und mit der Überprüfung durch die Normalformen wurde gezeigt, dass das Schema keine Redundanzen aufweist.

Ausbaufähigkeit besteht bzgl. der Szenarien. So soll es bspw. möglich sein, Daten innerhalb des Arbeitsplanes zu ändern, um Auswirkungen bzgl. der Berechnungsergebnisse und Szenarien sichtbar zu machen. Aber auch die Variierung der Produktionsmenge wäre eine zu betrachtende Ausgangslage.

Auf lange Sicht soll Quovadis erweitert werden, um innerhalb des Fraunhofer-IFF zur Anwendung zu kommen und die Dienstleistungserbringung der Fabrikplanung zu unterstützen. Als nächster Schritt soll die Erweiterung und die Datenbank implementiert werden, worauf wiederum die Anbindung an die VDT-Plattform aufbauen kann. Eine weitere Möglichkeit ist es Daten zu Personal-, Informations- sowie Ver- und Entsorgungsflüssen zu betrachten. Außerdem kann in das Werkzeug auch die Standortplanung einbezogen werden und um die Fälle des Rückbaus und der Revitalisierung erweitert werden.

Literaturverzeichnis

- [All99] Allgayer, F. (1999): Computerunterstützte Planung von Materialflusssystemen auf Basis statischer Materialflüsse. München.
- [AF09] Arnold, D.; Furmans, K. (2009): Materialfluss in Logistiksystemen. 6. Aufl., Berlin.
- [Dum03] Dumke, R. (2003): Software Engineering – Eine Einführung für Informatiker und Ingenieure: Systeme, Erfahrungen, Methoden, Tools. 4. Aufl., Wiesbaden.
- [EN02] Elmasri, R.; Navathe, S. (2002): Grundlagen von Datenbanksystemen. 3. Aufl., München.
- [End01] Enderle, J. (2001): XML in relationalen Datenbanken. Informatik-Spektrum, 24. Jg., Heft 6, S. 357-368.
- [DIN8580] DIN 8580 (1974): Fertigungsverfahren – Einteilung. Berlin.
- [GK07] Gienke, H.; Kämpf, R. (2007): Handbuch Produktion – Innovatives Produktionsmanagement: Organisation, Konzepte, Controlling. München.
- [Gru06] Grundig, C.-G. (2006): Fabrikplanung. 2. Aufl., München.
- [Hel10] Helbing, K. (2010): Handbuch Fabrikprojektierung. Heidelberg.
- [HL99] Hahn, D.; Laßmann, G.: Produktionswirtschaft – Controlling industrieller Produktion, Bd. 1 & Bd. 2. 3. Aufl., Heidelberg.
- [KE01] Kempler, A.; Eickler, A. (2001): Datenbanksysteme – Eine Einführung. 4. Aufl., Oldenbourg.
- [KSG84] Kettner, H.; Schmidt, J.; Greim, H.-R. (1984): Leitfaden der systematischen Fabrikplanung. München.
- [Kud07] Kudraß, T. (2007): Datenbanken. Leipzig.
- [LLS10] Laudon, K. C.; Laudon, J. P.; Schoder, D. (2010): Wirtschaftsinformatik – Eine Einführung. 2. Aufl., München.
- [LL07] Ludewig, J; Lichter, H. (2007): Software Engineering – Grundlagen, Menschen, Prozesse, Techniken. Heidelberg.

- [Poe08] Poenicke, O. (2008): Softwaregestützte Grobplanung und Visualisierung von Fabrikstrukturen im Automobilbereich. Diplomarbeit, Otto-von-Guericke-Universität. Magdeburg.
- [Rock 82] Rockstroh, W. (1982): Die technologische Betriebsprojektierung: Projektierung von Fertigungswerkstätten. Bd2. 2. Aufl., Berlin.
- [SSH08] Saake, G.; Sattler, K.-U.; Heuer, A. (2008): Datenbanken – Konzepte und Sprachen. 3. Aufl., Heidelberg.
- [SHS05] Saake, G.; Heuer, A.; Sattler, K.-U. (2005): Datenbanken: Implementierungstechniken. 2. Aufl., Bonn.
- [Sch10] Schwanz, A. (2010): Konzeptionelle Weiterentwicklung eines Werkzeugs zur softwaregestützten Grobplanung und Visualisierung von Fabrikstrukturen – Anforderungen, Möglichkeiten und Grenzen. Diplomarbeit, Hochschule für Technik und Wirtschaft. Dresden.
- [Sch70] Schmigalla, H. (1970): Methoden zur optimalen Maschinenanordnung. Berlin.
- [VDI5200] VDI-Richtlinie 5200 (2009): Fabrikplanung – Planungsvorgehen. Dortmund.
- [Voss08] Vossen, G. (2008): Datenmodelle, Datenbanksprachen und Datenbankmanagementsysteme. 5. Aufl., Oldenbourg.
- [Zäp01] Zäpfel, G. (2001): Grundzüge des Produktions- und Logistikmanagements. 2.Aufl., München.

Abschließende Erklärung

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

Magdeburg, den 17. November 2010