	<b>FUNKTIONALE WERKZEUGANFORDERUNGEN</b> Hauptteil Referenzmodell einer SEU	Seite 2-1
		Juni 1997

## 2 Referenzmodell einer SEU

### 2.1 Grobstruktur eines SEU-Referenzmodells

Im Rahmen des Entwicklungsstandards für IT-Systeme wird der Begriff „SEU“ (Softwareentwicklungsumgebung) wie folgt definiert:

Eine **SEU** ist die Gesamtheit aller technischen Betriebsmittel, die für die professionelle Entwicklung von IT-Systemen eingesetzt werden.

Betriebsmittel, aus denen sich eine konkrete SEU zusammensetzt, sind u. a. Hardware, Basissoftware (wie Betriebssysteme und Bürokommunikationssysteme) und vor allem Systementwicklungswerkzeuge (kurz „Werkzeuge“ genannt). Unter einem „Werkzeug“ für die Systementwicklung wird verstanden:

Ein **Werkzeug** ist ein Softwareprodukt, das die Erstellung oder Pflege/Änderung von IT-Systemen unterstützt.

Ein Werkzeug kann auch den Einsatz anderer Werkzeuge unterstützen (Beispiel: Entwicklungsdatenbank).

Eine SEU kann ihrerseits als ein IT-System aufgefaßt werden, das für den jeweiligen Nutzer<sup>1</sup> IT-Dienstleistungen bereitstellt.

Ein **Referenzmodell einer SEU** bringt alle IT-Dienstleistungen, die von einer SEU angeboten werden, in ein prinzipielles Ordnungsschema. Die fundamentalen Bausteine eines solchen Referenzmodells werden im folgenden **Leistungseinheiten** genannt. Die Leistungseinheiten des Referenzmodells einer SEU gliedern den Softwareanteil einer SEU aus Nutzersicht und bilden die Bezugselemente für die Festlegung von **Werkzeuganforderungen**. Dies geschieht dadurch, daß zu einer Leistungseinheit konkrete **Anforderungen an Werkzeuge** gestellt werden. Damit kann ein Werkzeug als Software verstanden werden, die die zu einer oder mehreren Leistungseinheiten gehörigen Anforderungen realisiert. Inhaltlich zusammengehörige Leistungseinheiten werden zu **Leistungskomplexen** zusammengefaßt.<sup>2</sup>

Das folgende Entity-Relationship (E/R)-Modell (Abbildung 2.1) zeigt den Zusammenhang zwischen Anforderung, Leistungseinheit und Werkzeug.

Eine zukunftsweisende SEU sollte

- sich dem Nutzer über alle ihre Werkzeugkomponenten hinweg durchgängig mit einer einheitlichen *Nutzeroberfläche* präsentieren;
- auf einer *Datenhaltung* basieren, mit der alle Informationen gespeichert werden können, die während der Systementwicklung entstehen. Die Datenhaltung sorgt für einen geregelten Informationsaustausch zwischen den einzelnen Werkzeugkomponenten. Außerdem werden hier alle zu speichernden Informationen einzeln und untereinander auf Konsistenz und Vollständigkeit überprüft. Die Datenhaltung speichert objektbezogene Informationen;

<sup>1</sup> Nutzer einer SEU sind diejenigen Personen, die an der Entwicklung oder Pflege/Änderung von IT-Systemen in unterschiedlichsten Rollen beteiligt sind. Alle diese Rollen sind im V-Modell definiert.

<sup>2</sup> Das Referenzmodell mit den Leistungseinheiten als fundamentale Bausteine stellt damit zunächst ein logisches Modell einer SEU dar, das über die Leistungseinheiten konkrete Anforderungen an Softwareprodukte (Werkzeuge) stellt.

- Komponenten zur Gewährleistung der *Sicherheit* (Vertraulichkeit und Integrität von Informationen und Verfügbarkeit von Informationen und Betriebsmitteln) besitzen;
- eine *Steuerungskomponente* besitzen, so daß die darin enthaltenen Werkzeuge auf einfache Weise miteinander kommunizieren können. Damit kann z. B. ein Werkzeug ein anderes zur Weiterverarbeitung der erzeugten Ergebnisse anstoßen<sup>1</sup>;
- weitestgehend den gesamten Leistungsumfang des *V-Modells* abdecken;
- eine *vertikale Integrität* der Einzelkomponenten gewährleisten, d. h. die erzeugten Ergebnisse einer Einzelkomponente müssen durch andere Einzelkomponenten weiter verwendet werden können.

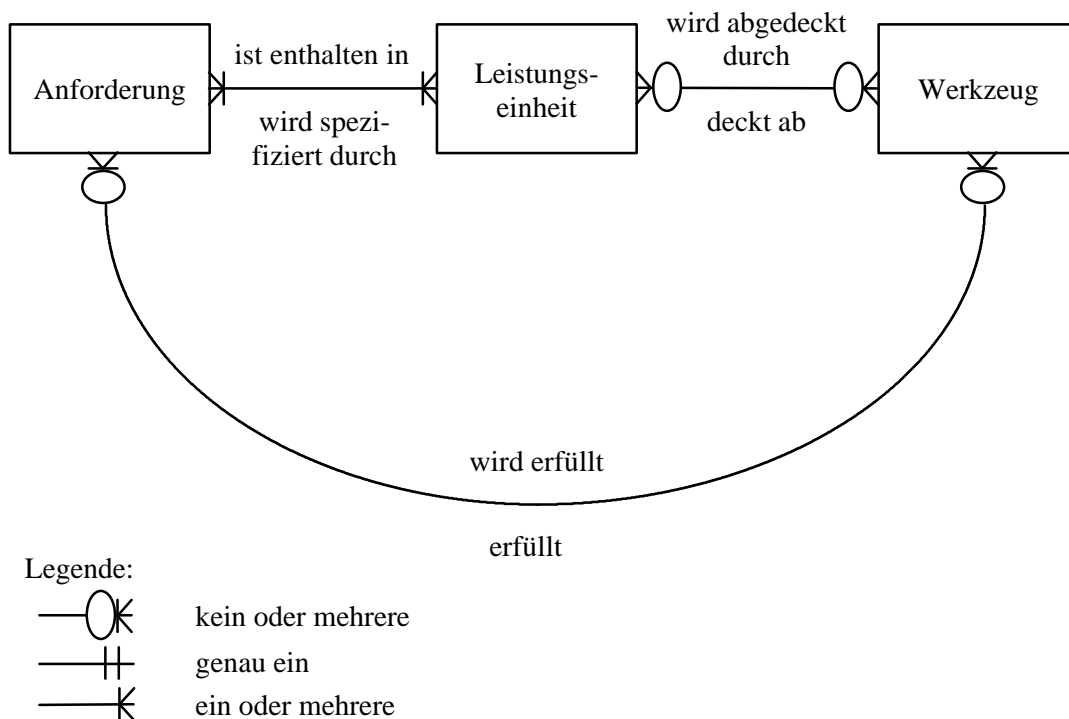


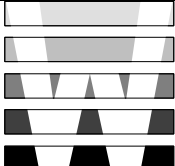
Abbildung 2.1: Zusammenhang von Anforderung, Werkzeug und Leistungseinheit

Die logisch-funktionale Gliederung der SEU in Leistungseinheiten verhindert eine voreilige Ausrichtung einer konkreten SEU auf bestimmte Merkmale, wie z. B.

- die Verteilung der Aufgaben auf Funktionsträger,
- die Verteilung der SEU-Leistungen auf Host- bzw. Arbeitsplatzrechner,
- die Konzeption existierender, herstellerabhängiger Werkzeugunterstützung,
- ein bestimmter, vordefinierter Stand an Sicherheit,
- ein vorgegebenes (anwendungsunabhängiges) Verständnis von Sicherheit,
- die Aufteilung der Sicherheitsfunktionalität auf bestimmte Sicherheitsmechanismen.

Im folgenden Abschnitt wird gezeigt, wie ein Referenzmodell für eine SEU unter Berücksichtigung obiger Anforderungen aussehen muß.

<sup>1</sup> Diese Forderung betrifft vor allem den Nachrichtenaustausch zwischen Werkzeugen in heterogenen Umgebungen und läuft auf die Erstellung eines entsprechenden Nachrichten-Standards hinaus. Da dies aber primär technische Details betrifft und bisher dazu lediglich noch nicht publizierte bei ANSI eingereichte Vorschläge verschiedener Hersteller existieren, wird im folgenden nicht näher darauf eingegangen.

	<b>FUNKTIONALE WERKZEUGANFORDERUNGEN</b> Hauptteil Referenzmodell einer SEU	Seite 2-3
		Juni 1997

## 2.2 Darstellung des Referenzmodells einer SEU

Im folgenden wird die dem V-Modell entsprechende SEU mit Hilfe eines Referenzmodells – kurz **SEU-Referenzmodell** genannt – dargestellt. Das SEU-Referenzmodell entspricht in der Gliederungsstruktur dem ECMA-Referenzmodell [1], ist aber an vielen Stellen detaillierter. Insbesondere werden im SEU-Referenzmodell die Anforderungen zur Systemerstellung, zur Qualitätssicherung, zum Konfigurations- und Projektmanagement im Einklang mit dem V-Modell mit Hilfe der Leistungseinheiten präzise formuliert. Im ECMA-Referenzmodell sind zu diesen Punkten entweder keine oder nur sehr vage Ausführungen zu finden. In einigen Teilen, wie z. B. den „Frameworks“ [1] weichen die beiden Referenzmodelle in Details voneinander ab. Ein detaillierter Vergleich zwischen dem hier mit Hilfe der Leistungseinheiten dargestellten SEU-Referenzmodell und dem ECMA-Referenzmodell findet sich im Anhang B.

Das SEU-Referenzmodell unterscheidet zwischen zwei Kategorien von Leistungseinheiten:

- den Leistungseinheiten, die im wesentlichen der Unterstützung der im V-Modell definierten Aktivitäten dienen. Sie betreffen unmittelbar die Arbeit der an der Entwicklung beteiligten Personen;
- denjenigen Leistungseinheiten, die um den im V-Modell definierten Entwicklungsprozeß herum allgemeine Hilfsfunktionen anbieten bzgl.
  - Objektverwaltung
  - Nutzerschnittstelle
  - Steuerung der Systementwicklung
  - Gewährleistung der Sicherheit.

Die hierzu gehörigen Leistungseinheiten entsprechen den „Frameworks“ im ECMA-Modell. Sie werden im folgenden als **Allgemeine Leistungseinheiten** bezeichnet.

Abbildung 2.2 zeigt eine grobe Struktur der SEU, wobei die Leistungseinheiten beider Kategorien zu **Leistungskomplexen** zusammengefaßt sind.

Die einzelnen Leistungskomplexe werden im folgenden kurz charakterisiert.

### 2.2.1 Allgemeine Leistungskomplexe

#### Objektverwaltung

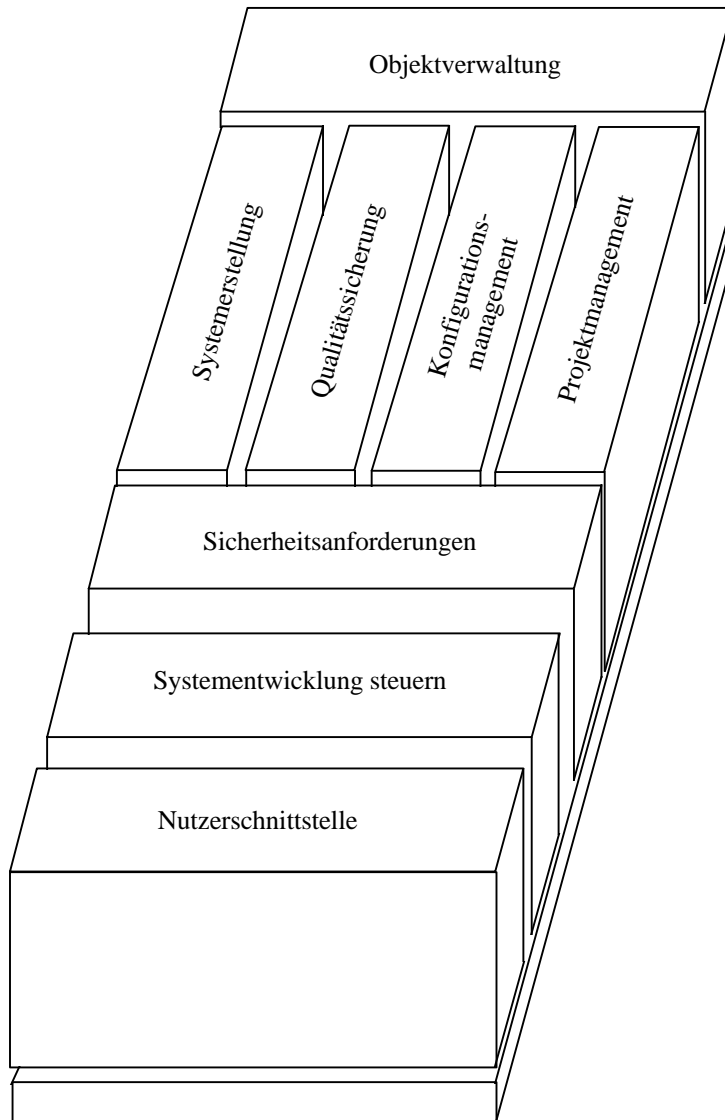
Der Leistungskomplex „Objektverwaltung“ definiert die Plattform,

- mit der alle Informationen über die Objekte, die während der Systementwicklung genutzt werden, in einem Datenmodell definiert, gespeichert und wieder aufgefunden werden,
- mit der innerhalb der SEU, d. h. zwischen den übrigen Leistungskomplexen, Informationen geregelt ausgetauscht werden und
- mit der Informationen in die SEU importiert und aus der SEU exportiert werden können.

Weiterhin gewährleistet die Objektverwaltung innerhalb der SEU:

- Mehrnutzerfähigkeit
- Transaktionsverwaltung

Die Objektverwaltung bestimmt ganz entscheidend den Integrationsgrad, den eine konkrete SEU erreichen kann.

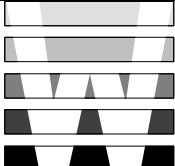


*Abbildung 2.2: Struktur des SEU-Referenzmodells*

### **Nutzerschnittstelle**

Hier werden folgende Leistungen zusammengefaßt:

- Meldungsgestaltung, Bedienerführung, Fenstertechnik und Skalierung
- Gestaltungsvarianten für Nutzerprofile
- Hilfefunktionen für den Nutzer
- Gestaltung von Dokumenten
- alphanumerische Bearbeitung von Dokumenten
- Bearbeitung von Grafiken.

	<b>FUNKTIONALE WERKZEUGANFORDERUNGEN</b> Hauptteil Referenzmodell einer SEU	Seite 2-5
		Juni 1997

### Systementwicklung steuern

Hier wird das Zusammenspiel aller Tätigkeiten des V-Modells gesteuert. Insbesondere regelt dieser Leistungskomplex die Anwendung aller Werkzeuge, die den Leistungskomplex „V-Modell“ realisieren. Damit werden den Nutzern der SEU kontextabhängig die übrigen Leistungseinheiten verfügbar gemacht.

### Sicherheit

Dieser Leistungskomplex umfaßt Funktionen:

- zur Gewährleistung der *Vertraulichkeit* von Informationen, d. h. zur Verhinderung eines unerwünschten lesenden Zugriffs zu Informationen, z. B. von nicht dazu autorisierten Personen,
- zur Gewährleistung der *Integrität* von Informationen, d. h. zur Verhinderung einer unerwünschten Modifikation von Informationen, z. B. durch nicht dazu autorisierte Personen,
- zur Gewährleistung der *Verfügbarkeit* von Informationen und SEU-Leistungen,
- zur Gewährleistung der *Verbindlichkeit* beim Nachrichtenaustausch.

Den Allgemeinen Leistungskomplexen des SEU-Referenzmodells kommt insbesondere dann eine Sonderstellung zu, wenn eine Integration der Werkzeuge verschiedener Hersteller in bezug auf

- eine einheitliche Datenhaltung (Objektverwaltung),
- eine einheitliche Nutzerschnittstelle,
- einen einheitlichen Austausch von Steuerungsinformation

erzielt werden soll.

## 2.2.2 Leistungskomplexe zur Unterstützung der V-Modell-Aktivitäten

### Systemerstellung

Dieser Leistungskomplex umfaßt alle Unterstützungsfunktionen zur Systemerstellung gemäß Submodell SE des V-Modells. Dies betrifft u. a. die Funktionsstrukturierung, die Prozeßmodellierung, die Informationsstrukturierung und die Generierung von Datenbanken.

### Qualitätssicherung

Hier werden die Leistungen zur statischen und dynamischen Prüfung von Produkten zusammengefaßt. Konstruktive Maßnahmen zur Qualitätssicherung sind in den anderen Leistungskomplexen (insbesondere im Leistungskomplex Systemerstellung) enthalten.

### Konfigurationsmanagement

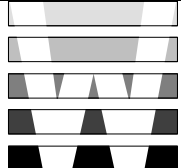
Der Leistungskomplex „Konfigurationsmanagement“ umfaßt Unterstützungsfunktionen zur Konfigurationsplanung und -kontrolle sowie zur Versionsverwaltung.

### Projektmanagement

Hier werden die Leistungen zu folgenden Aufgaben zusammengefaßt:

- Planung, Steuerung und Kontrolle der Systementwicklung,
- projektspezifische Verwaltung verfügbarer Ressourcen,
- Ausschreibungsrelevantes Tailoring des V-Modells.

Im SEU-Referenzmodell werden technische Elemente, die für den Betrieb einer rechnergestützten SEU erforderlich sind, wie z. B. Rechnerarchitektur, Netzwerksystem, Bürokommunikationstechnik, grundsätzlich nicht betrachtet. Die Hardwareeigenschaften erfahren keinerlei Berücksichtigung, da über das SEU-Referenzmodell rein funktionale Anforderungen gestellt werden.

Seite	2-6	<b>FUNKTIONALE WERKZEUGANFORDERUNGEN</b> Hauptteil Referenzmodell einer SEU	
	Juni 1997		

Ebensowenig werden übergreifende Dienstleistungen wie

- eine projektübergreifende Produktressourcenverwaltung,
- eine projektübergreifende strategische Informationsplanung und
- eine projektübergreifende Qualitätskontrolle

berücksichtigt.

## 2.3 Leistungseinheiten des SEU-Referenzmodells

Im folgenden Abschnitt 2.3.1 werden die Allgemeinen Leistungseinheiten und in Abschnitt 2.3.2 die Leistungseinheiten zum V-Modell tabellarisch zusammengefaßt. Die entsprechenden Detailbeschreibungen der Leistungseinheiten finden sich in Abschnitt 3.

### 2.3.1 Allgemeine Leistungseinheiten

<b>Objektverwaltung</b>	
LOBV01	Datenstrukturen und Daten der SEU erstellen, speichern, verwalten
LOBV02	Datenstrukturen und Daten der SEU auswerten, abfragen
LOBV03	Datenstrukturen und Daten der SEU exportieren, importieren

<b>Nutzerschnittstelle</b>	
LNSS01	Funktionen der Nutzeroberfläche
LNSS02	Nutzerprofil festlegen
LNSS03	Hilfefunktionen
LNSS04	Dokumentation aufbereiten
LNSS05	Texte editieren
LNSS06	Grafiken editieren

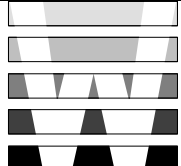
<b>Systementwicklung steuern</b>	
LSES01	Systementwicklung steuern

<b>Sicherheit</b>	
LSIC01	Allgemeine Sicherheitsprinzipien
LSIC02	Identifikation und Authentisierung
LSIC03	Zugriffskontrolle
LSIC04	Auditing
LSIC05	Wiederaufbereitung
LSIC06	Unverfälschtheit, Vertraulichkeit und Verbindlichkeit
LSIC07	Zuverlässigkeit der Dienstleistung

### 2.3.2 Leistungseinheiten der Leistungskomplexe zum V-Modell

<b>Systemerstellung</b>	
LSE01	Anforderungen erfassen
LSE02	Spezifikation von Nutzerschnittstellen unterstützen
LSE03	Architekturentwurf unterstützen
LSE04	Prozeßmodellierung unterstützen
LSE05	Funktionsstrukturierung unterstützen
LSE06	Spezifikation der Funktionen unterstützen
LSE07	Modellierung der Informationsflüsse unterstützen
LSE08	Modellierung der Funktionsdynamik unterstützen
LSE09	Informationsstrukturierung unterstützen
LSE10	Spezifikation der Komponenten und Module unterstützen
LSE11	Spezifikation der Datenbanken unterstützen
LSE12	Komponenten und Module generieren
LSE13	Datenbanken generieren
LSE14	Nutzerschnittstellenformate generieren
LSE15	Kompilieren
LSE16	Binden
LSE17	Debuggen
LSE18	Leistungen abschätzen
LSE19	Zeitverhalten simulieren
LSE20	Code rückwärts transformieren
LSE21	Datenbanken, Datenstrukturen rückwärts transformieren
LSE22	Klassen-/Objektmodellierung unterstützen
LSE23	Subsystemmodellierung unterstützen
LSE24	Moduldiagramme unterstützen
LSE25	Prozeßdiagramme unterstützen
LSE26	Use-Case-Modellierung unterstützen
LSE27	Zustandsmodellierung im objektorientierten Bereich unterstützen
LSE28	Interaktionsmodellierung unterstützen
LSE29	Formal spezifizieren
LSE30	Formal verifizieren
LSE31	Analyse verdeckter Kanäle

<b>Qualitätssicherung</b>	
LQS01	Prüfungen der Anforderungen unterstützen
LQS02	Prüfungen der Nutzerschnittstellen unterstützen
LQS03	Prüfungen der Architektur unterstützen
LQS04	Prüfungen der Prozeßmodelle unterstützen
LQS05	Prüfungen der Funktionsstrukturen unterstützen
LQS06	Prüfungen der Spezifikationen von Funktionen unterstützen
LQS07	Prüfungen der Informationsflußmodelle unterstützen
LQS08	Prüfungen der Ablaufmodelle unterstützen
LQS09	Prüfungen der Informationsstrukturen unterstützen

Seite	2-8	<b>FUNKTIONALE WERKZEUGANFORDERUNGEN</b> Hauptteil Referenzmodell einer SEU	
	Juni 1997		

LQS10	Prüfungen der Spezifikationen von Komponenten und Modulen unterstützen
LQS11	Prüfungen der logischen Schemaspezifikationen unterstützen
LQS12	Code statisch prüfen
LQS13	Texte statisch prüfen (Text ungleich Code)
LQS14	Prüffälle generieren
LQS15	Dynamische Prüfungen unterstützen
LQS16	Konsistenz des Ausschreibungsrelevanten Tailoring sichern
LQS17	Prüfungen der Projektressourcenverwaltung unterstützen
LQS18	Prüfungen der Projektorganisationspläne unterstützen
LQS19	Prüfungen der Projektablaufpläne unterstützen

<b>Konfigurationsmanagement</b>	
LKM01	Konfigurationsplanung und -kontrolle unterstützen
LKM02	Versionen verwalten

<b>Projektmanagement</b>	
LPM01	Projektressourcen verwalten
LPM02	Planung der Projektorganisation unterstützen
LPM03	Ausschreibungsrelevantes Tailoring unterstützen
LPM04	Planung des Projektablaufs unterstützen
LPM05	Kostenschätzung unterstützen
LPM06	Projektkontrolle und -steuerung unterstützen

### 2.3.3 Leistungseinheiten, Methoden und V-Modell-Aktivitäten

Methoden lassen sich vielfach erst dann effizient einsetzen, wenn eine ausreichende Werkzeugunterstützung zur Verfügung steht. Dies bedeutet, daß die Leistungseinheiten der Leistungskomplexe zum V-Modell Kriterien enthalten, die sich aus der Anwendung bestimmter Methoden ergeben. Die folgende Tabelle zeigt, welche Leistungseinheiten die in der Methodenzuordnung aufgeführten Methoden berücksichtigen.

Methoden	Leistungseinheit	
AUD	Audit	–
AVK	Analyse verdeckter Kanäle	LSE31
BALK	Balkenplan	LPM04
BAUM	Baumdiagramm	–
BBTE	Black-box-Testfallentwurf	LQS14
CRC	Class-Responsibility-Collaboration	–
DIAL	Dialog Design Modellierung	LSE02
DFM	Datenfluß-Modellierung	LSE07
DNAV	Data Navigation Modelling	LSE11
DVER	Designverifikation	LSE30
ELH	Entity Life History	LSE09
ER	E/R-Modellierung	LSE09
ETAB	Entscheidungstabellentechnik	–

<b>Methode</b>		<b>Leistungseinheit</b>
EVV	Earned Value Verfahren	LPM06
EXPM	Expertisemodell	–
FNET	Funktionsnetz-Modellierung	LSE08
FKTD	Funktionale Dekomposition	LSE05
FMEA	Failure Mode Effect Analysis	–
FS	Formale Spezifikation	LSE29
IAM	Interaktionsmodellierung	LSE28
KFM	Kontrollfluß-Modellierung	LSE07
KOM	Klassen-Objektmodellierung	LSE22
LOGM	Logische DB-Modellierung	LSE11
MODIAG	Moduldiagramme	LSE24
NORM	Normalisierung	LSE09
NPT	Netzplan-Technik	LPM04
NWA	Nutzwert-Analyse	–
OBJE	Objektentwurfstechnik	LSE03
OGG	Organigramm	LPM02
PCODE	Pseudocode	LSE10
PRODIAG	Prozeßdiagramme	LSE25
PVER	Programmverifikation	LSE30
PZIM	Prozeßinteraktionsmodellierung	LSE04
REV	Review	–
SIMU	Simulationsmodelle	LSE19
SMOD	Schätzmodelle	LPM05
SSM	Subsystemmodellierung	LSE23
STAT	Statische Analyse	LQS01–LQS13
STRD	Structured Design	LSE03
SVM	Systemverhaltensmodelle	LSE08
T	Testen	LQS15
TRDA	Trend-Analyse	LPM06
UCM	Use-Case-Modellierung	LSE26
WBTE	White-box-Testfallentwurf	LQS14
ZUST	Zustandsübergangsmmodellierung	LSE04
ZUSTO	Zustandsmodellierung im objektorientierten Bereich	LSE27
ZUVM	Zuverlässigkeitsmodelle	–

Die folgenden Fälle lassen sich unterscheiden:

1. Eine Leistungseinheit deckt genau eine Methode ab. Ein Beispiel für diesen Fall ist die Leistungseinheit LSE05 „Funktionsstrukturierung unterstützen“, die ausschließlich die Unterstützung der Methode Funktionale Dekomposition (FKTD) zum Ziel hat.
2. Eine Leistungseinheit deckt mehrere Methoden ab. Dies ist immer dann sinnvoll, wenn die Methoden eine zusammenhängende Aufgabenstellung unterstützen. Ein Beispiel für diesen Fall ist die Leistungseinheit LPM04 „Planung des Projektablaufs unterstützen“, die die Methoden Balkenplan (BALK) und Netzplan-Technik (NPT) unterstützt.

3. Eine Methode wird durch mehrere Leistungseinheiten abgedeckt. Dieser Fall tritt bei der Methode Statische Analyse (STAT) auf, da hier die Methode auf unterschiedliche Prüfgegenstände angewendet werden muß.
4. Eine Methode wird durch keine Leistungseinheit abgedeckt. Ein Beispiel für diesen Fall ist die Methode Review (REV), bei der eine Werkzeugunterstützung nicht möglich ist.
5. Eine Leistungseinheit, die eine Methode abdeckt, enthält gegebenenfalls noch weitere Anforderungen, die nicht methodisch bedingt sind. Ein Beispiel für diesen Fall ist die Leistungseinheit LSE02 „Spezifikation von Nutzerschnittstellen unterstützen“, die neben der Unterstützung der Methode „Dialog Design Modellierung (DIAL)“ beispielsweise noch nicht methodisch bedingte Kriterien zur Spezifikation fensterorientierter Nutzerschnittstellen enthält.
6. Vielfach ist jedoch auch für die Aktivitäten des V-Modells eine Werkzeugunterstützung sinnvoll, für die die Methodenzuordnung keine Methoden vorsieht. Demzufolge gibt es auch eine Reihe von Leistungseinheiten, die keine Methoden unterstützen. Ein Beispiel hierfür ist die Leistungseinheit LSE17 „Debuggen“.

Der Zusammenhang zwischen Leistungseinheiten, Methoden und V-Modell-Aktivitäten ist in Abbildung 2.3 illustriert. Dabei wurden die oben angeführten Fälle eingezeichnet.

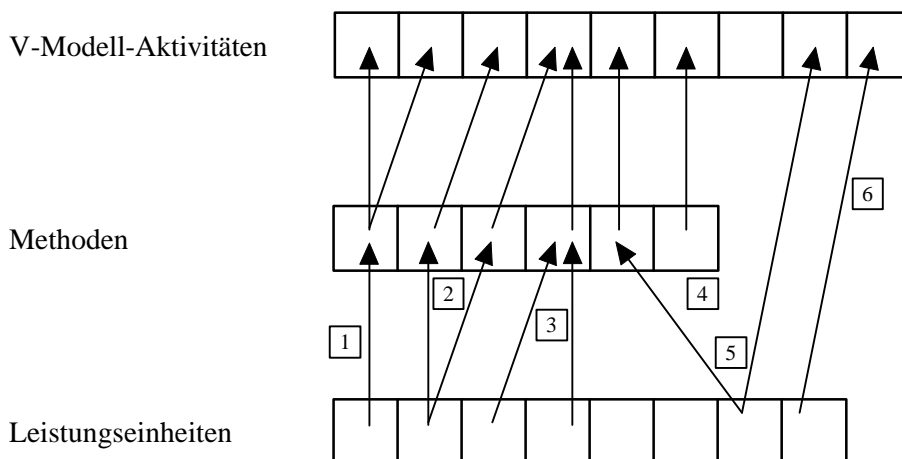


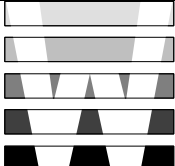
Abbildung 2.3: Zusammenhang der Leistungseinheiten, Methoden und Aktivitäten

### 2.3.4 Leistungseinheiten und Werkzeuge

Um eine konkrete SEU einzurichten, müssen die Anforderungen der Leistungseinheiten durch konkrete Werkzeuge erfüllt werden. Ein Werkzeug, das integraler Bestandteil einer konkreten SEU sein soll, muß mindestens Anforderungen einer, kann aber durchaus auch die Anforderungen mehrerer Leistungseinheiten erfüllen. Die Erwartung, daß die Abdeckung aller Leistungseinheiten durch ein einziges Werkzeug möglich ist, ist nicht realistisch.

### 2.3.5 Leistungseinheiten und Varianten

Bestimmte Dienstleistungen der SEU können in unterschiedlichen Ausprägungen erforderlich sein. Ein typisches Beispiel hierfür ist die Leistungseinheit „Binden“, für die bei der Auswahl einer konkreten SEU je eine Ausprägung (Variante) für jede zu bearbeitende Programmiersprache erforderlich ist. Es ist hier offensichtlich, daß jedes für diese Leistungseinheit in Frage kommende Werkzeug nur eine Variante der Leistungseinheit abdeckt.

	<b>FUNKTIONALE WERKZEUGANFORDERUNGEN</b> Hauptteil Referenzmodell einer SEU	Seite 2-11
		Juni 1997

Varianten sind nur dort vorzusehen, wo eine vollkommen identische (semantische) Funktionalität für Eingaben/Ausgaben mit unterschiedlicher Syntax erforderlich ist. Im einzelnen sind Varianten bei folgenden Leistungseinheiten zu berücksichtigen:

- LNSS05 „Texte editieren“:  
Varianten sind abhängig von der Syntax der zu editierenden Texte (z. B. Programmiersprachen-Syntax, Prosa-Syntax)
- LNSS06 „Grafiken editieren“:  
Varianten sind abhängig von der Syntax der zu editierenden Grafiken
- LSE12 „Komponenten und Module generieren“:  
Varianten sind abhängig von der Zielsprache
- LSE13 „Datenbanken generieren“:  
Varianten sind abhängig von dem zu verwendenden Datenbanksystem
- LSE14 „Nutzerschnittstellenformate generieren“:  
Varianten sind abhängig vom TP-Monitor der Zielumgebung
- LSE15 „Kompilieren“:  
Varianten sind abhängig von der zu verarbeitenden Programmiersprache
- LSE16 „Binden“:  
Varianten sind abhängig von der Syntax des Objektcodes und vom Zielsystem
- LSE17 „Debuggen“:  
Varianten sind abhängig von der zu verarbeitenden Programmiersprache
- LQS13 „Texte statisch prüfen (Text ungleich Code)“:  
Varianten sind abhängig von der Syntax des zu prüfenden Textes (z. B. Prosa, formale Spezifikationstexte)
- LQS14 „Prüffälle generieren“:  
Varianten sind abhängig von der Syntax des Prüfgegenstandes
- LQS15 „Dynamische Prüfungen unterstützen“:  
Varianten sind abhängig von der Syntax des Prüfgegenstandes

### 2.3.6 Leistungseinheiten von SE und QS

Einige Leistungseinheiten des Leistungskomplexes SE stehen in sehr engem Verhältnis zu ihren korrespondierenden Leistungseinheiten des Leistungskomplexes QS, z. B. LSE05 „Funktionsstrukturierung unterstützen“ und LQS05 „Prüfungen der Funktionsstrukturen unterstützen“. Da in der Praxis viele Aktivitäten der QS bereits im Bereich der Systemerstellung durchgeführt werden, lassen sich die Anforderungen an die beiden Leistungseinheiten nicht immer gut trennen, zumal auch viele Werkzeuge beide Aspekte berücksichtigen.

Seite 2-12	<p style="text-align: center;">FUNKTIONALE WERKZEUGANFORDERUNGEN Hauptteil Referenzmodell einer SEU</p>	
Juni 1997		

### 2.3.7 Beziehungen von Leistungseinheiten des Leistungskomplexes Sicherheit zu Leistungseinheiten anderer Leistungskomplexe

Einige Leistungseinheiten des Leistungskomplexes Sicherheit stehen in sehr engem Verhältnis zu Leistungseinheiten aus anderen Leistungskomplexen, z. B. LSIC03 „Zugriffskontrolle“ mit den Anforderungen an Zugriffsarten zu LOBV01 „Datenstrukturen und Daten der SEU erstellen, speichern, verwalten“ mit den Anforderungen an Zugriffe. Die betreffenden Anforderungen lassen sich daher nicht immer eindeutig nach einzelnen Leistungseinheiten trennen, zumal sie in der Praxis normalerweise durch ein und dasselbe Werkzeug abgedeckt werden müssen.

### 2.3.8 Hinweise zur Beschreibung der Leistungseinheiten

In Abschnitt 3 werden die in den Abschnitten 2.3.1 und 2.3.2 aufgeführten Leistungseinheiten nach einem einheitlichen Schema beschrieben. Dieses Schema ist wie folgt aufgebaut:

#### 1. Zuordnung zum V-Modell und zur Methodenzuordnung

Hier werden jeweils die (Teil-) Produkte der (Teil-) Aktivitäten des V-Modells aufgeführt, die mit der Leistungseinheit unterstützt werden. Sind für diese (Teil-) Aktivitäten in der Methodenzuordnung Elementarmethoden vorgeschrieben, so werden diese hier ebenfalls aufgeführt, da die Leistungseinheiten dann diese Methoden unterstützen müssen.

Die Zuordnung erfolgt in einer einheitlichen Struktur:

Kurzbezeichnung der (Teil-) Aktivität/Langbezeichnung

(Teil-) Produkt

Methode

#### **Beispiel:**

##### *V-Modell-Aktivität (-Produkt)*

- SE 3.3 Anforderungen an die Funktionalität definieren („Technische Anforderungen.Gesamtfunktion“)
- SE 4.1-SW SW-Architektur entwerfen („SW-Architektur“)

##### *Methode*

- Prozeßinteraktionsmodellierung (PZIM)
- Zustandsübergangsmodellierung (ZUST)

#### 2. Kurzcharakteristik

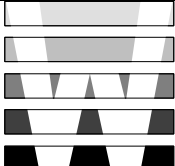
Hier wird jeweils der Leistungsumfang einer bestimmten Leistungseinheit verbal erläutert.

#### 3. Anforderungen

##### 3.1 Anforderungen an die Schnittstellen

Besondere Anforderungen an die Nutzerschnittstelle<sup>1</sup>, Zusammenspiel mit anderen Leistungseinheiten, zu erfüllende Konsistenzbedingungen

<sup>1</sup> Generelle Anforderungen an die Nutzerschnittstelle sind der Leistungseinheit LNSS01 „Funktionen der Nutzeroberfläche“ zugeordnet.

	FUNKTIONALE WERKZEUGANFORDERUNGEN Hauptteil Referenzmodell einer SEU	Seite 2-13
		Juni 1997

### 3.2 Anforderungen an die Methodenunterstützung

Anforderungen, die sich aus der Anwendung bestimmter Methoden ergeben

### 3.3 Anforderungen an die Funktionen

Anforderungen an die vom Werkzeug zur Verfügung gestellten nicht methodisch bedingten Funktionen

### 3.4 Sonstige Anforderungen

Anforderungen, die nicht in eine der obigen Kategorien einzuordnen sind