

CAD-Frameworks: Die Probleme jenseits der Entwurfswerkzeuge

Klaus Quibeldey-Cirkel

Der Framework-Technologie wird eine maßgebliche Rolle in der Wachstumsentwicklung des EDA-Marktes im allgemeinen und der ASIC-Industrie im besonderen zugesprochen. Um den Produkt-Entwicklungszyklus weiter zu straffen, werden Soft- und Hardware am Ende der 90er Jahre ausschließlich in standardisierten CAD-Infrastrukturen entworfen. Kürzere Markteinführungszeiten sind nicht mehr allein durch leistungsgesteigerte Werkzeuge möglich, sondern nur noch im standardisierten Verbund aller Entwurfskomponenten: Benutzerführung, Methoden, Werkzeuge und Datenmodelle. Der Artikel reflektiert die Framework-Konzepte, die im BMFT-Forschungsprojekt DASSY entwickelt wurden (DASSY: „Datentransfer und Schnittstellen für offene integrierte VLSI-Entwurfssysteme“).

Die W-Fragen

Das Chip-Zeitalter der Höchstintegration hat ein kaum zu überschauendes CAx-Szenario hervorgebracht, treffend umschrieben durch die Metapher *Islands of Automation*. CA-Entwurfsmethoden für kundenspezifische Schaltungen und mannigfaltige CA-Techniken der Entwurfssynthese und -verifikation bilden ein komplexes Instrumentarium. Mit Einführung der *simulierten Netzliste* als formaler Schnittstelle zwischen Chip-Entwerfer und ASIC-Hersteller sind Entwurf und Test in die alleinige Verantwortung des Entwicklungsingenieurs übergegangen. Einerseits liegt die erweiterte Kompetenz in seinem Sinne: als Vorzüge der vorgezogenen Netzlisten-Schnittstelle seien hier nur die verkürzte *Turn-around-Zeit* und die höhere Entwurfssicherheit genannt. Andererseits werden hohe Anforderungen an sein methodisches Systemdenken gestellt: „Welche Entwurfswerkzeuge von welchem Hersteller setze ich wann und in welchem Kontext ein? Wie sind die Schnitt-

stellen definiert? Wie sind die Ein- und Ausgabedaten zu interpretieren, d. h. auf welchen Modellbildungen beruhen sie?“ An diese W-Fragen knüpft sich weiter die Forderung nach der konsistenten Verwaltung der Mega-Daten von Entwurf und Test.

Was fehlt ist eine pragmatische, in einer standardisierten Umgebung eingebettete Entwurfsmethodik, die seit der *Introduction to VLSI Systems* von Carver Mead und Lynn Conway vor einem Jahrzehnt erst in jüngster Zeit in neuen Ansätzen verfolgt wird. So hat sich zum Beispiel das Firmenkonsortium CFI (CAD Framework Initiative) zum Ziel gesetzt, bislang isolierte Werkzeuge herstellerübergreifend mit Hilfe eines prozeduralen Schnittstellenmodells zu integrieren; so stehen die massiven Modellierungsaktivitäten in der EDIF-Gemeinde unter dem Anspruch, den Informationsaustausch zwischen EDA-Werkzeugen und Entwurfsstätten zu standardisieren [4]; und so zeichnet sich ein neuer Trend zum *design consulting* mittels wissensbasierter Systeme zur Modellierung von Fachkompetenz ab.

VLSI-Infrastrukturen

Integrationssysteme, neudeutsch *Frameworks* oder gehoben „(Software)-Infrastrukturen“, machen nur dann einen wirtschaftlichen Sinn, wenn die Dienstleistungen und Schnittstellen Standards genügen, die von der Industrie aktiv unterstützt werden. Zwar haben die Schnittstellenprobleme einen neuen Markt für Softwarehäuser geschaffen, die „maßgeschneiderte“ Einzellösungen offerieren, für den EDA-Markt insgesamt aber zeichnet sich ein Sättigungstrend ab: EDA-Anwender investieren zurückhaltender in neue Produkte, die mit eigenen *In-house*-Systemen oder den Werkzeugen von Drittherstellern unverträglich sind oder aber hohe Anpassungskosten mit sich

<i>Trend</i>	<i>1. Generation</i> <i>70er Jahre</i>	<i>2. Generation</i> <i>80er Jahre</i>	<i>3. Generation</i> <i>90er Jahre</i>
<i>Standardisierung</i>	gering: firmeneigene Datenformate, erste singuläre, nur für bestimmte Werkzeugarten definierte Standards (CIF)	wachsend: hersteller-neutrale Datenformate (EDIF), Schnittstelle zwischen ASIC-Entwickler und -Hersteller: simulierte Netzliste	dominierend: Datenzugriff und -austausch weitestgehend standardisiert (EDIF, VHDL, CFI)
<i>Werkzeug-Integration</i>	isolierte, nicht kompatible Werkzeuge (Insellösungen, <i>In-house</i> -Systeme)	geschlossene, herstellerspezifische <i>Turn-key</i> -Systemlösungen	<i>Frameworks</i> : offene Infrastrukturen für heterogene kooperierende Werkzeuge
<i>treibende Kräfte</i>	Werkzeugentwickler vorwiegend aus universitären Forschungseinrichtungen	CAD/CAE-Anwender, ASIC-Technologien	EDA-Industrie, Normungsgremien, Vielfalt an Hardware-Plattformen und Betriebssystemen

Tabelle 1: Generationswechsel der Hardware-Entwurfsumgebungen

bringen.

Die internationalen Bemühungen um langlebige Industrienormen im EDA-Bereich sollen dabei keineswegs die Vielfalt der Produkte schmälern. Im Gegenteil: Die Stagnation des EDA-Marktes und im Gefolge die der ASIC-Industrie soll überwunden werden, indem das Spektrum an alten und neuen Werkzeugen verschiedener Hersteller breitbandig nutzbar gemacht wird. Das Schlüsselwort heißt: *Kompatibilität*. Die Entwicklung des EDA-Marktes ist geprägt durch die enge Kopplung zwischen dem Integrationsgrad der Entwurfswerkzeuge und dem Grad der Standardisierung ihrer Schnittstellen. Der Generationswechsel in der Werkzeugkonfiguration für den industriellen Hardware-Entwurf belegt diesen Trend: Tabelle 1.

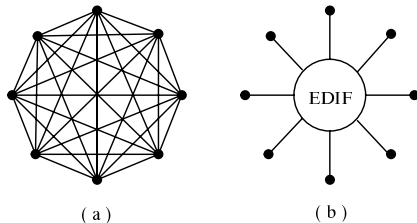


Bild 1: Schnittstellenvielfalt: $O(n^2)$ versus $O(n)$

Bild 1 verdeutlicht die Schnittstellenproblematik in heterogenen CAD/CAE-Entwurfsumgebungen am Beispiel des elektronischen Datenaustauschformats EDIF. Die Daten- und Steuerflüsse zwischen den einzelnen Entwurfswerkzeugen bzw. Entwicklungsstätten (dargestellt als n Knoten) sind im allgemeinen bi-direktional. Die gegenwärtig noch dominierenden Insellösungen lassen folglich den Aufwand für die Kommunikation (Datenkonverter und Tool-Interfaces) quadratisch mit der Vielfalt der Werkzeuge anwachsen: Situation (a). Allein standardisierte Schnittstellen können die Aufwandskosten für kompatible Umgebungen drastisch reduzieren: Situation (b).

Schnittstellen

Mit ihrer massiven Unterstützung für normierte CAD-Infrastrukturen verfolgen die EDA-Anbieter und -Anwender eine von der Phono/Video-Industrie inspirierte *Plug-in-and-play*-Philosophie, die sich ihrem Kompatibilitätsanspruch gemäß folgendes zum Ziel gesetzt hat:

- Austausch und Wiederverwendung gemeinsamer Entwurfsdaten

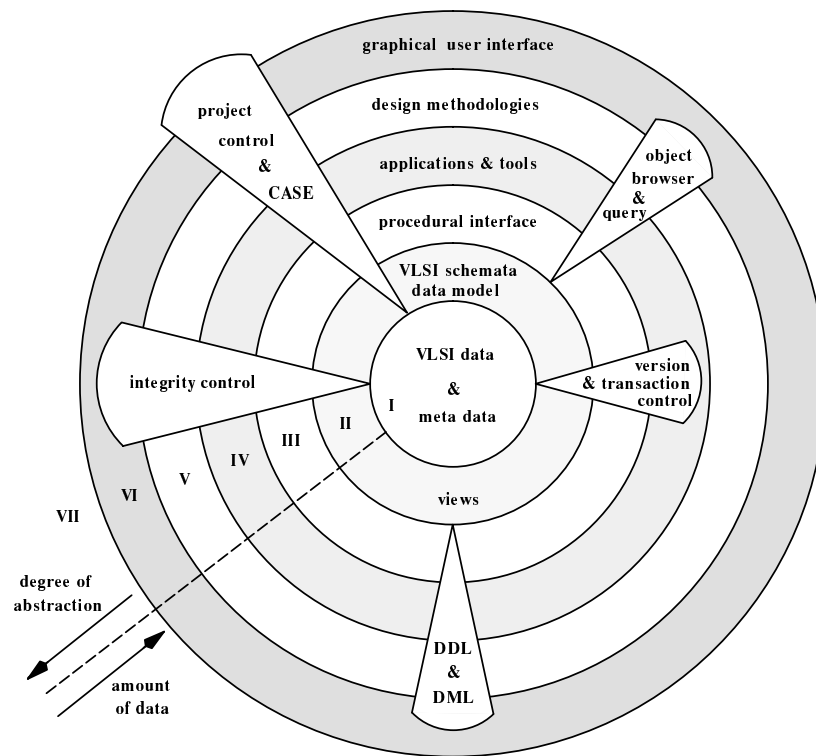


Bild 2: Schalenmodell einer VLSI/CAD-Infrastruktur

- Einheitliche Benutzerführung und Methodenkontrolle
- Rechnerunterstütztes Prozeß- und Ressourcen-Management

Bild 2 zeigt das Schalenmodell einer idealisierten VLSI/CAD-Infrastruktur. Die Darstellung als isolierende Schichten entspricht der Intention, die Entwurfskomponenten (Benutzerführung, Methoden, Werkzeuge und Daten) klar voneinander zu trennen und ausschließlich kontrollierte Interaktionen über die Schnittstellen zuzulassen.

Zwei gegenläufige Achsen durchziehen die Schnittstellen: Mit zunehmender Abstrahierung nimmt der Detaillierungsgrad der betrachteten Entwurfsdaten ab, verringert sich also das Datenaufkommen. Umgekehrt geht die Konkretisierung der Entwurfsdaten mit einem größeren Aufwand für die Datenhaltung einher. Analog verhält es sich mit der Entwurfskomplexität: Je weiter „außen“ der Entwickler den Entwurfsprozeß steuert und seine

Entscheidungen trifft, desto geringer ist seine kognitive Belastung durch die Details.

Die Schnittstellen im Überblick:

I–II: VLSI-Rohdaten und Meta-Daten (Daten über Rohdaten) werden gemäß dem Datenbankschema organisiert und zentral verwaltet. Direkte Zugriffsmöglichkeiten bestehen nicht.

II–III: Datenbankschemata und -teilschemata (*views*), erstellt auf der Grundlage eines objektorientierten Datenmodells, strukturieren die Objekt- und Beziehungstypen des abzubildenden Weltabschnitts.

III–IV: Die prozedurale Werkzeugschnittstelle fungiert als Bindeglied zwischen der statischen Datenhaltung und den dynamischen Einwirkungen der Entwurfswerkzeuge auf den Datenbestand.

IV–V: Das Tool-Management verwaltet die Entwurfswerkzeuge gemäß der gewählten

Entwurfsmethodik: Auswahl der Werkzeuge, Generieren von Steuerdateien und Aufrufsequenzen etc.

V-VI: Das Design-Management überwacht den Entwurfsfluß, d. h. die zeitliche Abfolge synthetisierender und verifizierender Prozesse, und koordiniert den Input/Output-Datenfluß an den Schnittstellen der einzelnen Entwurfsabschnitte.

VI-VII: Die grafische Benutzeroberfläche schließlich überbrückt die interaktive Schnittstelle Mensch-Maschine, die nach software-ergonomischen Gesichtspunkten gestaltet ist.

Schnittstellen sollen „wohldefinierte“ und normative Übergänge schaffen und zugleich für übergreifende Interaktionen durchlässig sein. Wir können folgende übergeordnete Dienstleistungen durch mehrere Schnittstellen hindurch unterscheiden:

▷ *Integritätskontrolle* I-VI: Die Überwachung der Datenintegrität erstreckt sich über alle internen Schnittstellen einer CAD-Infrastruktur und erfaßt somit die statischen, dynamischen und methodischen Entwurfsaspekte. Eine ausgewogene Kombination von zentralen und verteilten Kontrollmechanismen sorgt für die Einhaltung der Integritätsbedingungen.

▷ *Projektkontrolle* I-VII: Dem Systementwerfer steht die Möglichkeit offen, jederzeit Status- und Prozeßinformation abzufragen und auszuwerten. Das Tool- und Design-Management vergleicht die aktuellen Konstruktionsdaten mit der Projektspezifikation und schlägt korrigierende Entwurfsentscheidungen vor (Expertensysteme) oder nimmt diese vom Entwerfer entgegen. Zur Verwaltung aller logistischer Daten stehen entsprechende CASE-Werkzeuge zur Verfügung.

▷ *Versions- und Transaktionsverwaltung* I-IV: Die für den VLSI-Entwurf charakteristischen Anforderungen an ein Versions- und Transaktionskonzept — Protokollierung der

zeitlichen und alternativen Entwurfsschritte und Teilergebnisse, Synchronisierung konkurrierender Datenzugriffe — resultieren aus der arbeitsteiligen Durchführung eines Projekts im Team. Auswahl und Zusammenstellung der für den Gesamtentwurf relevanten Versionen — die Konfiguration der Einzelentwürfe — stehen unter der Projektkontrolle.

▷ *Datenbeschreibung und -manipulation* II-VI: Für die Definition der Datenbank-schemata und der Zugriffs- und Verwaltungsoperationen verfügt der Datenbank-Administrator über die von konventionellen Datenbanken bekannten Sprachmittel: DDL und DML (*Data Definition/Manipulation Language*). In VLSI-Infrastrukturen aber orientieren sich diese Sprachmittel nicht an einfachen Datenstrukturen wie Records und Tabellen, sondern sind auf die semantisch mächtigeren Organisationsprinzipien des Objekt-Paradigmas ausgerichtet: Klassen, Objekte, Beziehungen und Constraints.

▷ *Objekt-Browser und Datenabfrage* II-VI: Zur Visualisierung des aktuellen Datenbestands (Objektebene) und der den Daten zugrundeliegenden Organisationsstrukturen (Schema-Ebene) werden von der grafischen Schnittstelle aus Objekt- und Schema-Browser eingesetzt. Weiterhin stehen die aus der Datenbanktechnologie bekannten prädikatenlogischen Abfragetechniken zur Verfügung, um beispielsweise datensensitive Entwurfsentscheidungen abzusetzen oder Objektmengen nach gemeinsamen Eigenschaften zu qualifizieren.

Derzeit gibt es eine Vielzahl verschiedener VLSI-Frameworks. Sie weichen in einigen wichtigen Aspekten von der Idealvorstellung des Schalenmodells ab und stellen oft nur herstellereigenspezifische Integrationsrahmen für die eigene Produktlinie dar. Um aber innovative Entwurfswerkzeuge in einen bestehenden Werkzeugverbund aufnehmen zu können (*technology push*) oder mit einem neuen Produkt schnell am Markt zu sein (*market pull*), bedarf es herstellerneutraler Dienstleistungs-Standards.

Projektlandschaft

Daß Frameworks nicht nur ein marketing-wirksames Schlagwort sind, sondern auch einen zentralen Gegenstand der Praktischen und Technischen Informatik darstellen, zeigt die aktuelle Forschungslandschaft. Schätzungen des amerikanischen Instituts für Verteidigungsanalysen (IDA) zufolge, verringern einheitliche Entwurfsumgebungen die Produkt-Entwicklungszeiten um 40 bis 60 Prozent. Das Bostoner Marktforschungsinstitut Technology Research Group (TRG) taxiert den Produktivitätsgewinn auf 30 Prozent [1]. Die aktuelle internationale CAD-Projektlandschaft wurde in erster Linie durch das Wissen um den ökonomischen Faktor „Framework“ motiviert. Die wichtigsten Framework-Projekte und -Standardisierungsbemühungen lassen sich zwanglos den einzelnen Schnittstellen im Schalenmodell zuordnen.

Auf dem Weg zu industrieweit anerkannten CAD-Standards leisten die skizzierten Framework-Projekte und Normungsgremien die entscheidende Arbeit und bieten zugleich die Foren für EDA-Produktanbieter und -anwender, um die Entwicklung und Etablierung der Framework-Technologie aktiv voranzutreiben.

VLSI-Weltmodelle

Eine effiziente Kommunikation zwischen heterogenen Werkzeugen erreicht man nur durch Integration und nicht allein durch Import/Export-Dienste auf Dateibasis. Um aber Entwurfswerkzeuge integrieren zu können, bedarf es vor allem einer *schematisierten Modellvorstellung* über den Informationspool, der während des Entwurfsprozesses auf allen Darstellungsebenen angereichert wird und der das Entwurfsobjekt mit seiner Entstehungsgeschichte vollständig zum aktuellen Zeitpunkt beschreibt.

Die Generalanforderungen an die Modellierung eines derartigen Informationspools nennen wir „Weltmodell“ in Anlehnung an den Begriff des „Datenmodells“ in der Datenbanktechnologie. Wie Datenmodelle (zum

Beispiel das relationale) stellen Weltmodelle die konzeptionellen Werkzeuge zur Modellierung eines interessierenden Weltausschnitts zur Verfügung. Im Mittelpunkt der datenintensiven Framework-Technologie stehen also zwei Facetten der Informationsverarbeitung:

1. Modellieren und Schematisieren der Entwurfsinformation: Datenrepräsentation
2. Konsistentes Fortschreiben und Abfragen des Datenbestands: Datenverwaltung

Damit einher gehen zwei funktionale Aspekte: Werkzeug- und Datenaustausch-Schnittstellen. Erstere sind prozedural und im Zugriff on-line. Von der Konzeption her stellen sie abstrakte Datentypen dar: Das Werkzeug kennt nur die Operatoren zum Abfragen und Manipulieren des Datenbestands. Wie der Datenzugriff organisiert ist, wie beispielsweise die Datenstrukturen oder der Mechanismus, um die Objektidentität zu gewährleisten, realisiert sind, bleibt dem Werkzeug verborgen. Datenaustausch-Schnittstellen dagegen sind deskriptiv und arbeiten in der Regel off-line: Der Datentransfer erfolgt geschlossen in einer formatierten ASCII-Datei. EDIF und VHDL sind hier die Standardformate.

Um das Weltmodell einer Applikation aufzustellen, müssen wir zunächst deren statischen Strukturen erfassen — eine Momentaufnahme des Informationsgehalts untersuchen. Zum statischen Anteil eines Weltmodells gelangen wir über die in den letzten Jahren aufgekommene formale Informationsmodellierung.

Informationsmodellierung

Ursprünglich als formale Sprache für die Definition eines allgemeinen Produktmodells entwickelt, avancierte das STEP-Produkt EXPRESS innerhalb weniger Jahre zum Standard-Medium der *Informationsmodellierung* in CFI und anderen Framework-Initiativen. EXPRESS knüpft unmittelbar an die Methoden und Techniken der semantischen Datenmodellierung an. Sie ist eine normative objektorientierte Modellierungssprache — textuell und grafisch — und forma-

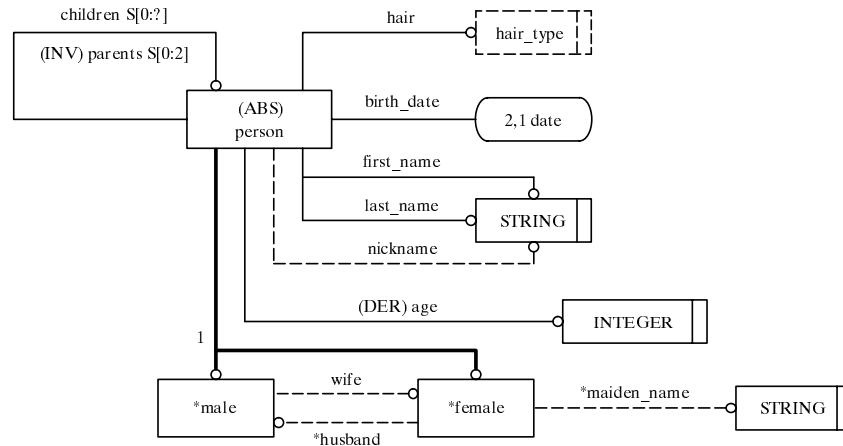


Bild 3: EXPRESS-G-Beispiel

liert (a) die Beschreibung der Objekte eines Diskursbereichs (*universe of discourse*) sowie deren Beziehungen untereinander und (b) die Beschränkungen (*constraints*), denen die strukturellen Eigenschaften unterliegen.

Die Gründe zur Informationsmodellierung im Framework-Bereich sind zwingend:

- Konsensfindung unter Experten verschiedener Entwurfsbereiche über den Bedeutungsinhalt der Entwurfsdaten: normative Entwurfssemantik
- Kommunikationsmittel, um ein Problem zu analysieren und eine Lösung zu spezifizieren: Teamunterstützung
- Grundlage für die Implementierung als Datenaustauschformat und Datenbank-schema: Referenzmodell
- konzeptioneller Unterbau von CAD-Frameworks und deren Schnittstellen

Bild 3 gibt ein illustratives Beispiel für die Informationsmodellierung in EXPRESS-G, der grafischen Untermenge von EXPRESS [5]. Das gezeigte Modell beschreibt das konzeptionelle Konstrukt Person. Eine Person besitzt gewisse Eigenschaften, wie Vor- und Zunamen, optional einen Rufnamen, Haarfarbe,

Geburtsdatum und daraus abgeleitetes (Derived) Alter. Eine Person ist eine abstrakte (ABSTRACT) Oberklasse in dem Sinne, daß es kein direktes Exemplar von ihr gibt. Erst die weitere Spezialisierung von Person führt auf „instanzierbare“ Unterklassen, nämlich männliche und weibliche Personen. Zwischen letzteren können monogame Beziehungen bestehen, was dann zusätzliche Eigenschaften in der Namensführung der Eheleute mit sich bringt (die Modellierung des aktuellen deutschen Namenrechts dürfte sich aufwendiger gestalten). Eine Person kann natürlich Kinder haben, die wiederum Personen mit obigen Eigenschaften und Beziehungen sind. Der Rollename der Eltern/Kind-Beziehung gibt Auskunft über die jeweilige Sichtweise.

Daß einige Eigenschaften bestimmten (nur in der textuellen Form von EXPRESS sichtbaren) Beschränkungen unterliegen, wird durch einen vorgestellten Asterisk (*) vermerkt. Einheiten, die eine Menge von konzeptionellen oder realen Objekten mit gleichen Eigenschaften repräsentieren, werden in EXPRESS-G durch Rechtecke gekennzeichnet. Beziehungen sind durch Linien zwischen den Rechtecken erkennbar. Strichstärke und Strichart stehen dabei für Art und Stärke der Beziehung.

Datentransferformate

Die Notwendigkeit, die Syntax-Inflation einzudämmen — den Informationsgehalt der Entwurfsdaten zu modellieren bevor die Form der Daten festgelegt wird —, wurde in den internationalen Standardisierungsgremien erst in letzter Zeit erkannt. Im Bestreben, Form und Inhalt, also Syntax und Semantik, nicht zu vermischen, nimmt das Austauschformat für elektronische Entwurfsdaten EDIF eine Vorreiterrolle ein. Um die von vielen Anwendern prognostizierte Syntax-Krise zu überwinden, hat das EDIF Technical Committee bereits 1989 entschieden, daß zukünftige Erweiterungen des Standards nur noch auf der Grundlage eines wohlfundierten Informationsmodells zugelassen werden. Als formales Beschreibungsmittel wurde hierfür EXPRESS ausgewählt.

Ende 1989 wurde die „EDIF European TSC Information Modelling Working Group“ (kurz: EDIF-IMG) ins Leben gerufen, die sich ausschließlich mit der Informationsmodellierung der EDIF-Domäne befaßt. CAE-Entwurfsdaten werden ansatzweise auch in anderen Gremien und Projekten modelliert: *ECAD Domain Model* (EIS-Projekt), *Conceptual Model* (ECIP) und *Test Model* (EVEREST). Die textuelle und grafische Gesamtdokumentation der EDIF-Informationsmodelle kann [4] entnommen werden.

Die Motivation für die Informationsmodellierung von EDA-Austauschformaten läßt sich wie folgt zusammenfassen:

- Sie führt auf ein syntax-unabhängiges Modell der durch das Format bezeichneten CAD-Objekte,
- formalisiert die Zusammenhänge und Restriktionen in der Objektwelt,
- spielt beispielsweise die Rolle eines Referenzmodells, das den inhaltlichen Vergleich zweier formatierter Datenbeschreibungen erlaubt, und
- sie sollte als Ausgangspunkt und Grundlage für die Syntax-Definition dienen.

In diesem Sinne nützt die Informationsmodellierung im Kontext des Datenaustauschs sowohl den Entwicklern als auch den Anwendern der Framework-Technologie.

Prozedurale Schnittstellen

Wir haben bereits darauf hingewiesen: In einem Framework erfolgt die Schnittstellenanpassung nicht allein dadurch, daß Entwurfsdaten als *blobs* (*binary large objects*) transferiert werden. Vielmehr muß auch die Möglichkeit bestehen, diese Daten einzeln oder im interessierenden Zusammenhang zu interpretieren und zu manipulieren. Mit anderen Worten: Entwurfswerkzeuge sollen über die Framework-Schnittstellen miteinander operieren und gegebenenfalls austauschbar sein, um so die Integrationskosten gering zu halten. Prozedurale Werkzeugschnittstellen leisten das Gewünschte. In Anlehnung an die Terminologie der Datenbankler sprechen wir hier von einem *Datenmodell*, das es für prozedurale Interfaces zu entwerfen gilt. Die Aufgabenstellung ist ähnlich wie bei der geschilderten Informationsmodellierung für Datenaustauschformate, nur daß hier der *dynamische* Aspekt des Datenzugriffs hinzukommt.

Prozedurale Werkzeugschnittstellen und ihre konzeptionelle Modellierung sind Brennpunkte der Entwicklungsaktivitäten in CFI und DASSY. Das DASSY-Datenmodell (*DaDaMo*) ist ein effizienter Konzept-Mix aus der semantischen und objektorientierten Datenmodellierung. Seine Implementierbarkeit wurde bereits prototypisch nachgewiesen [3] und seine Relevanz für die Framework-Standardisierung international anerkannt. *DaDaMo* beinhaltet erweiterte Konzepte für die Definition und Manipulation komplexer Entwurfsdaten.

Entwurfsdaten, wie Schaltungskomponenten und Verbindungsstrukturen, werden grundsätzlich als *Objekte* im Sinne des Objekt-Paradigmas modelliert (Datenkapselung, Klassenzugehörigkeit) und gehandhabt (abstrakte Datentypen, klassenbezogene Methoden). Die statischen Eigenschaften eines Objekts werden durch seine Attribute, die strukturellen durch seine (attribuierbaren) Beziehun-

gen zu anderen Objekten und die dynamischen durch benutzer-definierte Operationen beschrieben. Die Definition von Constraints erlaubt darüber hinaus, die Objekteigenschaften individuell einzuschränken, um den besonderen Anforderungen der Entwurfswerkzeuge zu genügen. DaDaMo unterstützt die Modellierung von VLSI-spezifischen Weltausschnitten, indem es die multiple Vererbung der Objekteigenschaften zuläßt und die Beziehungstypen frei definierbar hält. Welche Semantik eine Beziehung trägt — sowohl deskriptiv als auch operativ (Rollenfunktion) — und welche Objektstrukturen darstellbar sind — vernetzt oder verschachtelt —, bestimmt der Anwender.

Die operative Semantik der Beziehungen wird *aktiv* unterstützt. Im Gegensatz zu anderen Vorschlägen drückt DaDaMo Beziehungen nicht allein durch die Angabe von Objektreferenzen aus, verbirgt das, was auf den Beziehungen geschehen soll, nicht in unzugänglichem Operationscode oder beläßt es gar auf bloße Hinweise durch entsprechende Bezeichner. Beziehungstypen, die oft im Mikroelektronik-Entwurf benötigt werden, sind in DaDaMo bereits vordefiniert. Über die voreingestellten Beziehungstypen können weitere definiert werden. Diese leiten sich stets aus den vordefinierten ab und werden durch benutzereigene Methoden in ihrer operativen Semantik weiter differenziert. DaDaMo bietet nach außen zwei Sprachschalen für die Schemabeschreibung an. Zum einen steht eine EXPRESS-orientierte Schale zur Verfügung, um die in EXPRESS geschriebenen Informationsmodelle nahtlos in die Werkzeugschnittstelle zu integrieren, und zum andern eine an C++ angelehnte Sprachschale, da der Großteil aller industriellen Framework-Applikationen in C bzw. C++ geschrieben ist.

Die Weltmodelle der Framework-Applikationen umfassen also den statischen Informationsgehalt einer Objektmenge, vermehrt um den dynamischen Zusatz aus der konzeptionellen Modellierung der Datenhaltungs- und Werkzeugschnittstellen. Diese Weltmodelle stammen nicht nur aus der Feder des Systemanalytikers, sondern haben auch ihre Gültigkeit und praktische Bedeutung für die Entwurfs- und Implementierungsebenen.

Ausblick

Die nicht-algorithmischen Probleme mit immensem Datenaufkommen stehen heute — zu Beginn der 90er Jahre — im Brennpunkt der Informatik. Frameworks für Soft- und Hardwareumgebungen stellen eine derartige Problemklasse dar. Der Handlungsbedarf zielt auf die Problembeschreibung durch den Endbenutzer und die Problemanalyse durch den Systementwickler. Das hier angesiedelte moderne *Requirements Engineering* und *Prototyping* setzt den unproblematischen Anschluß an das traditionelle Software-Engineering voraus.

Die konzeptionelle Modellierung bietet hierfür eine tragfähige Bindung und ein gemeinsames Sprachmittel. Die gemeinsamen Abstraktionstechniken aus der Künstlichen Intelligenz, der objektorientierten Programmiermethodik und der semantischen Datenmodellierung erlauben es, das Wissen über einen interessierenden Weltausschnitt geeignet zu repräsentieren und effektiv für die Verarbeitung aufzubereiten. Zum Aufstellen der Weltmodelle stehen bereits mächtige Konzepte und Techniken zur Verfügung. Grafik-Tools wie Generatoren, Editoren und Browser, Management-Tools für die Versionierung und Konfiguration von Objektmengen und Syntax-Checker erleichtern dem Weltmodellierer das Leben. Konstruktion, Dokumentation und formale Analyse seiner Modelle erfolgen bereits auf dem Rechner.

Was wir brauchen, sind Code erzeugende Werkzeuge, die das konzeptionelle Modell „zum Laufen bringen“, d. h. in die Konstrukte konventioneller Hochsprachen bzw. für VLSI-Applikationen in eine Hardware-Beschreibungssprache wie VHDL compilieren. Das sind zum Beispiel Schema-Compiler, die ein Weltmodell automatisch in ein Datenbankschema überführen, oder Modellsimulatoren, die eine Aussage über die Konsistenz der gespeicherten Objekte und der Interaktionen auf ihnen erlauben. Das heißt also: Rechnerunterstützung nicht nur für die textuelle und grafische Darstellung von Weltmodellen, sondern auch für die Inspektion und Implementierung dieser. Die Probleme jenseits der Werkzeuge sind bekannt, harren aber noch ihrer systematischen Lösung durch stan-

standardisierte Framework-Modelle.

Literatur

- [1] *Bhat, J.; Taku, F.*: Frameworks: EDA-Technologie für die 90er Jahre. „Elektronik“ 39 (1990) H. 21, S. 88–94
- [2] *DASSY-Projekt*: DaDaMo: The DASSY Data Model. GMD, St. Augustin, 1992
- [3] *DASSY-Projekt*: The DASSY Prototype. GMD, St. Augustin, 1992
- [4] *EDIF-IMG*: Information Model of EDIF Version 2.0.0. Wilkes, W. (Hrsg.), FernUniversität Hagen, 1992
- [5] *Schenk, D.*: EXPRESS Language Reference Manual. McDonnell Aircraft Company, St. Louis, MO., 1990