

Database Model for Zero Defect Quality

Peter Czerner, Dr. Ralf Montino
Elmos Semiconductor AG
Dortmund

Schlüsselworte

Zero Defect, Halbleiter, relationales Datenbankmodell

Einleitung

Die Komplexität eines Halbleiterchips, sein Design, Prozesse, hunderte Produktionsschritte und nicht zuletzt die Ablage der Testdaten in einer Datenbank ist eine Kunst für sich. Wird zusätzlich eine Zero Defect Qualität verlangt, ist eine kosteneffiziente Halbleiterherstellung nur auf der Grundlage einer hochqualitativen Datenverwaltung möglich.

Kernanforderung für ein System ist neben praktisch 100%iger Verfügbarkeit die Konsistenz der abgelegten Daten. Die Verfügbarkeitsanforderungen werden durch ein entsprechend stabiles Produkt mit weiteren Enterprise Features wie Online-Backup erfüllt. Basisanforderungen wie Multiuser Unterstützung, Performance, Langzeitstabilität des Produktes und des Anbieters, Rechtesteuerung werden selbstverständlich vorausgesetzt.

Ein Relationales Datenbanksystem - wie das Oracle RDBMS - erfüllt diese Anforderungen. Es ermöglicht über die reine Datenspeicherung weit hinausgehende Prozessabbildungen konsistent und betriebssicher zur Verfügung zu stellen.

Die Elmos Semiconductor AG stellt kunden- und applikationsspezifische Halbleiterchips und Sensoren her. Seit fast 2 Jahrzehnten bilden unsere Experten das Wissen und die Abläufe des Unternehmens in ein zentrales relationales Datenbankmodell ab. Es erfasst sowohl das gesamte Produktionsleitsystem, die Verwaltung der Bauelemente, die unzähligen Messwerte und Messrezepte, Lagerverwaltung wie auch die Abbildung des Produktentwicklungsprozesses. Das Model steuert 3 Fabriken im Produktionsverbund ebenso wie die Entwicklung, Produktion und Auslieferung unserer Produkte.

Das Konzept des wachsenden zentralistischen Datenmodells ist die Grundlage für die Konsistenz, Verfügbarkeit und die Qualität unserer Daten. Andererseits verbirgt es die nicht vernachlässigbaren Nebeneffekte, wie einen höherer Verwaltungsaufwand, den Verlust an Flexibilität und die Kosten der notwendigen Hochverfügbarkeit.

Dieser Beitrag beschreibt unser Konzept eines zentralen Datenbankmodells in einem Mittelstandunternehmen auf Basis von Oracle RDBMS.

Rezept für ein Halbleiter oder: Wie produziert man ICs mit Hilfe eines RDBMS?

Allein das Design eines integrierten Schaltkreises (englisch integrated circuit, kurz IC) ist ohne hochkomplexe Software Entwicklungswerkzeuge nicht denkbar. Auch der Herstellungsprozess muss in jedem Detail beherrschbar sein, sei es, wenn es um nanometergenaue Beschichtung mittels einer tonnenschweren Plasmaanlage oder um die minutengenaue Logistik im Reinraum geht. Der Herstellungsprozess eines ICs beinhaltet viele hundert Produktions- und Prüfschritte und mehrere Testschritte bei unterschiedlichen Umweltbedingungen. Eine zentrale Steuerung von mehreren Fabriken, die Verwaltung von Fertigungsabläufen und Datenerfassung für eine große Palette an

Produkten geht nur mit Hilfe eines komplexen wie effizienten Produktionsleitsystem (englisch: Manufacturing Execution System, MES).

Nahezu 500 Spezialisten arbeiten alleine in der Fertigung der ELMOS Semiconductor AG in Dortmund daran, kundenspezifische ICs und Sensoren für den sehr anspruchsvollen Automotiv Markt herzustellen. Unser Qualitätsstandard hat einen einfachen Namen, Zero Defect (ZD) und ein der Werkzeuge dafür ist das auf einem Oracle RDMBS basierendes MES.

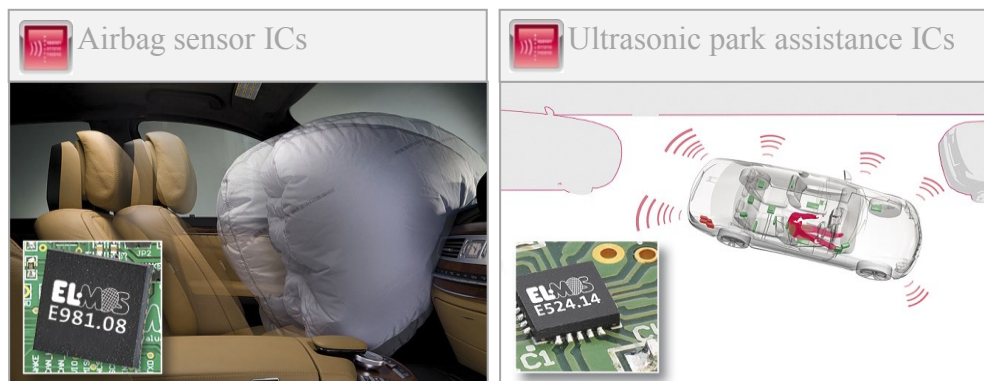


Abb. 1: Sicherheitsrelevante Anwendungen brauchen Zero Defect Qualitätstandard, hier am Beispiel von Airbag Sensor und Park Assistance IC [1]

Zero Defect: Fehlervermeidung als Qualitätsstandard

Zero Defect (ZD) wurde in den 60er Jahren des letzten Jahrhunderts in der Amerikanischen Luft- und Raumfahrtindustrie mit dem konkreten Ziel entwickelt, die Fehler bei der Produktion der Pershingrakete zu reduzieren [2]. Das amerikanische Verteidigungsministerium griff die Idee 1964 auf und setzte sich seit dem aktiv für die Umsetzung des ZD Programms bei den Waffenlieferanten [3] ein. Die Idee war einfach wie effektiv: Man fand heraus, dass die übliche Fehlertoleranz in der Produktion dazu führt, dass viele vermeidbare Fehler passieren und unnötige Qualitätskosten verursachen. Wird dagegen fehlerfreie Produktion in jedem Teilprozess propagiert oder gar verlangt, fühlen sich die Prozessbeteiligten dazu motiviert, Fehler zu vermeiden. In den 70er Jahren machte Philip Crosby, für Pershing verantwortlicher Qualitätsmanager bei Martin Company, die Idee in seinen zahlreichen Publikationen populär. Im Buch „Quality is Free“ [4] vertrat er die Meinung, dass durch ZD induzierte Qualitätskostenreduktion das größte Potenzial für den Kostenreduktion eines Unternehmens darstellt. Im Klartext: Im ständigen Wettbewerb muss man Qualität produzieren, sie lässt sich nicht kosteneffektiv durch Test und Selektion erreichen. Das ist das praktische und schlagende Argument für ZD.

Etwa 30 Jahre nach dem ersten ZD Programm wurde in den 90er Jahren im Zuge der Aufteilung der Produktionsschritte auf zahlreiche Zulieferer die Idee von der Automobilindustrie aufgegriffen und umgesetzt. Auf diesem Wege kam sie zu uns: ZD wird bei ELMOS Semiconductor AG umgesetzt und gelebt.

Datenqualität als Grundbaustein für ZD

Die Komplexität der Produkte – nicht nur in der Halbleiterproduktion – steigt von Tag zu Tag. Sie wird durch die steigende Anzahl an Komponenten zusätzlich verstärkt. Auch die Gleichteilestrategie

im Allgemeinen - die Anzahl der Standardkomponenten in den Produkten wächst kontinuierlich – erhöht den potenziellen Impact im Fehlerfall: Mehr verschiedene Produkte werden durch eine fehlerhafte Komponente oder einen fehlerhaften Prozess betroffen.

Steigende Produktkomplexität lässt nur mit intelligenten rechnerunterstützten Systemen handhaben. Sie ermöglichen weitest gehende Automatisierung, kontinuierliche Überwachung der Prozesse und Anlagen und die nötige Rückverfolgbarkeit des Materials. Die Voraussetzung ist eine eindeutige Beschreibung und Abbildung der Prozesse. Dieses gilt sowohl der Herstellungs- wie auch der Geschäfts-Prozesse und es klingt viel leichter, als es ist.

Wo die Datenverarbeitung und rechnerunterstützte Produktionslenkung feste Bestandteile des Produktionsprozesses sind, ist deren Qualität maßgeblich an der Qualität der Produkte beteiligt. So führt z.B. ein Fehler im elektronischen Prozessrezept zur Fehlprozessierung und damit oft zur Verschrottung der Ware. Ähnlich können falsch zugeordneten Messwerte zur Dejustage im darauf folgenden Prozess und im Endeffekt zur Fehlfunktion führen. Man könnte hier unzählige Beispiele für die feste Verkettung der IT Prozesse mit dem Herstellungsprozess und deren kausalen Zusammenhang mit der Fehlerquote der hergestellten Produkte anführen. Die Folgerung ist klar: Wo ZD ein Qualitätsstandard in der Produktion ist gilt er auch für die IT Systeme.

Relationales Datenbankmodell mit Oracle

Der Griff zur modernen Datenbanktechnologie liegt nahe. Dabei ist eine Datenbank heutzutage nicht nur eine Datenschenke, auch wenn sie oft nur als solche benutzt wird. Ein RDBMS bildet vielmehr die Beziehungen der Daten ab und sorgt für deren Konsistenz. Es ist die Methode zur Abbildung der Prozesse.

Schon 1995 wurde eine Oracle RDBMS bei ELMOS Semiconductor AG als zentrales Datenbanksystem eingesetzt, damals in der Version 7.2.2. Dafür sprachen schon damals die breite Schnittstellenpalette die Oracle bot und natürlich die Reife der Software in Bezug auf Kernfunktionalität, Systemsicherheit und Stabilität.

Mit Ansi SQL, Constraints, Trigger und einer integrierten prozeduralen Programmiersprache bietet Oracle eine solide Grundlage für die Abbildung eines Unternehmens in Datenstrukturen. Die Vielzahl an Schnittstellen ermöglicht weitgehend direkte Ankopplung unterschiedlichster Geschäftslogik- und Produktions-Systeme. Die Zentralisierung des Datenmodells und Datenbestandes ist der nächste logische Schritt.

Abbildung ohne Systembruch: Ein Modell für das ganze Unternehmen

Die Verfügbarkeit der Schnittstellen zu einer Oracle Datenbank ist das erste, wenn auch nicht das wichtigste Argument für ein zentrales Datenbankmodell. Nehmen wir an, es gibt 10 unterschiedliche Softwarewerkzeuge, die Datenaustauschen müssen. Im ungünstigsten Fall, wo jedes Werkzeug mit jedem kommunizieren muss, brauchen wir 88 Input/Output Schnittstellen. Nicht mitgerechnet sind evtl. Verbindungen zu lokalen Datenbanken.

Wenn man allerdings ein zentrales Austauschpunkt für Daten einrichtet und alle Softwarewerkzeuge über diesen Austauschpunkt kommunizieren und Daten austauschen lässt, reduziert sich die Anzahl der Schnittstellen auf 10! Dieses Modell ist natürlich sehr vereinfacht, trotzdem als erste Näherung ausreichend. Wieso also nicht eine Datenbank als Knotenpunkt des Datenflusses setzen? Die Struktur wird dadurch deutlich einfacher und die einzige Anforderung an die Schnittstellen jedes neuen Werkzeuges lautet: Es braucht eine Oracle Anbindung...

Hinzu kommt: Es ist schon eine erhebliche Herausforderung, die Datenkonsistenz innerhalb eines Datenmodelles wirklich sicher zu stellen. Über verschiedene Produkte von verschiedenen Herstellern hinweg halten wir das für nicht erreichbar.

Je mehr unternehmenswichtige Daten in einer Datenbank gesammelt sind, umso einfacher ist es, deren realen Beziehungen in einem Datenbankmodell abzubilden. Hier wird das wichtigste Argument für eine zentrale Datenbank deutlich: Wir können Bereich-, Prozess- und Zweck-übergreifende Datenbeziehungen abbilden. Praktisch bedeutet es, dass die Konsistenz der Daten zentral mit

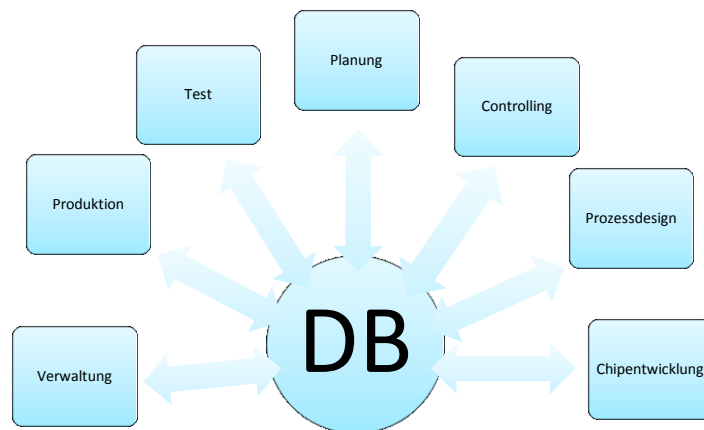


Abb. 2: Zentrales Datenbankmodell vereinfacht die Datenflüsse und reduziert die Anzahl der nötigen Schnittstellen

Datenbankmittel überwacht werden kann. Auch in punkto Redundanz gibt es einen großen Vorteil verglichen mit der sonst nötigen Synchronisierung der Daten und den Fehlern, die durch die inhaltlichen Diskrepanzen der Datenquellen entstehen.

Ein zentrales Datenbankmodell bedeutet auch eine zentrale Datenquelle für alle möglichen Berichte: Der Zugang zu aktuellen Informationen wird einfacher, kann im Umkehrschluss aber auch einfacher kontrolliert werden: Auch hier sind die internen Datenbankmethoden zur Stelle.

Nebenbei werden Kosten für Hochverfügbarkeit und Administrationsaufwände reduziert: Ein willkommener Nebeneffekt.

Ein zentrales Datenbankmodell kann also vier Aspekte der ZD Datenqualität - **Redundanz, Konsistenz, Sicherheit und Verfügbarkeit** - gewährleisten.

Und die Nachteile? Natürlich gibt es auch diese. Zum einen ist der Integrationsaufwand neuer Anwendungen größer, als wenn sie lokal und unabhängig von anderen Bereichen und Anwendungen erstellt würden. Zum anderen ist durch die Vernetzung der Daten der Austausch und das Entfernen von veralteten Datenstrukturen nicht ohne weiteres möglich: Hier muss man zuerst alle Abhängigkeiten finden und auflösen. Nicht zuletzt ist der Zukauf und Integration von Drittanbietersoftware schwierig: Meist bringt jeder Anbieter seine eigene Datenbank mit. Diese muss dann aufwändig mittels Schnittstellen an das zentrale Modell angekoppelt werden. Im Optimalfall pflegt die Anwendung ihre Daten in einem separaten Schema derselben Datenbankinstanz und der Datenaustausch geschieht mittels Datenbankmethoden innerhalb der Datenbank. Die Frage des Führenden Systems ist allerdings in jedem Einzelfall sorgfältig aufzuklären.

Entwurf und Entwicklung, Aufbau und Wachstum: Struktur ist kein Nachteil

Bevor Sie hingehen und versuchen auf einen Schlag ein Datenbankmodell für Ihr gesamtes Unternehmen aufzubauen, zunächst eine Warnung: Bevor sie fertig sind, wird ihr Model veraltet sein. Wir stellten unsere Datenhaltung schrittweise auf Oracle um. Zuerst entstanden die Knotenpunkte des Modells, Strukturen mit den wichtigsten Daten des Unternehmens, z.B. Produkte, Prozesse, Mitarbeiter, Anlagen etc. Danach wurden unzählige lokale Tabellen und Anwendungsdatenbanken zentralisiert und verknüpft. Dabei wuchs das Datenbankmodell und drum herum wurden die Zugriffswerkzeuge und Schnittstellen umgestellt und neue erzeugt. Dieser Prozess dauert praktisch bis heute an, auch wenn die Änderungsfrequenz des Modells mit dem initialen Wachstum der 90er Jahre nicht vergleichbar ist. Das Datenbankmodell wächst und ändert sich entsprechend der Prozesse des Unternehmens. D.h. Im Umkehrschluss, dass die Flexibilität des Modells wichtiger ist, als seine Vollständigkeit. Die Paradigmen und Qualitätsregeln beim Aufbau des Modells und überhaupt bei der Softwareerstellung waren fortan:

- Zentralisierung
- Sicherstellung der Datenkonsistenz über Datenbankmechanismen und nicht über die Applikation
- Auf die Produktion optimierte Zugriffszeit
- Nach Möglichkeit ewige Datenhaltung (wegen Nachweispflicht)
- Qualitätssicherung der Datenhaltung und Softwareerstellung durch Reviews und Freigaben

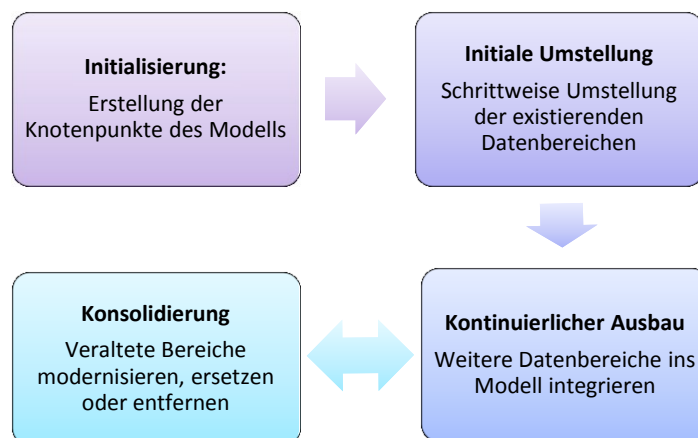


Abb. 3: Entwicklungszyklus für das Datenbankmodell

Methoden und Werkzeuge

Unsere Methoden zum Model- und Software-Entwicklung wurden mit den Jahren natürlich verändert und an die wachsenden Anforderungen angepasst. Nicht alles ist an der Stelle fest vorgeschrieben, um Freiheiten für Kreativität offen zu lassen. Folgendes lässt sich festhalten:

Entwurf und Visualisierung

Alle Datenbankobjekte und deren Änderungen werden einfachheitshalber in Skriptform archiviert und per SVN versioniert. Jedes Objekt ist zwingend per Tabellenkommentare zu dokumentieren. Der Entwurf selbst entsteht und wird visualisiert mittels PL/SQL Developer ®.

Wir versuchen schon lange nicht mehr, das Datenbankmodell als ganzes visuell darzustellen und das aus zwei einfachen Gründen: Zum einen ist es sehr aufwändig, das Abbild des Modells zu pflegen oder neu zu erstellen. Zum anderen trägt ein Gesamtbild wenig zum Verständnis des Modells bei, wo schon unser Hauptschema mit seinen tausend Tabellen eine etwa 10m² Papierfläche brauchen würde, würde man es ausdrucken.

Produktionsstrukturen ins Datenbankmodell überführen

Beim Modellentwurf sind immer drei Parteien am Tisch: Der Prozessverantwortliche („Kunde“), der Softwareentwickler und der Datenbankadministrator. Jeder von Ihnen repräsentiert seine Sicht des Prozesses und alle sind für die Entstehung des Modells und der darauf basierenden Software unentbehrlich.

Überwachung muss auch mal sein...

Trotz aller künstlerischen Freiheit in der Softwareentwicklung, die ihrerseits für Motivation und Ideenvielfalt sorgt, funktioniert Qualitätssicherung nicht ohne Überwachung. Unsere Methoden sind hier

- Fest definierte Freigabeverfahren für Software und Modelentwicklung
- Monitoring
- Messung der Modelqualität

Schwarze Löcher sind kontraproduktiv: Datenzugang für alle!

Durch die zentralisierte Datenhaltung vereinfacht sich der Datenzugriff immens, vorausgesetzt, jeder findet die Daten, die er sucht. Das ist allerdings durch den Dokumentationszwang der Datenbankobjekte sowie die Nähe zu den Entwicklern gewährleistet. Und wenn keiner der unzähligen dynamisch erzeugten Berichte das liefert, was man sucht, wird mittels SQL eine geeignete Abfrage erzeugt, auch in den IT fernen Bereichen. Die SQL Schulungen sind bei uns sehr gefragt!

Hochverfügbarkeit

Durch die tiefe Integration der Geschäftsprozesse mit den Prozessleitsystemen führt eine Fehl- oder Nichtfunktion der zentralen Instanz zum faktisch sofortigen Stillstand der Fertigung und zu sehr ernsthafter Behinderung praktisch aller Bereiche des Konzerns. Die sich daraus ergebende Anforderung wird durch eine Vielzahl von Maßnahmen und mehrheitlich selbstentwickelten Werkzeugen erfüllt. Wir erreichen seit vielen Jahren eine Verfügbarkeit von mehr als 99,99%. Wichtig ist hier, dass die geplanten Nichtverfügbarkeiten (z.B. Upgrades) enthalten sind. Einen Cluster (Betriebssystem/Applikation/Hardware) setzen wir hier nach mehrfacher sorgfältiger Prüfung nicht ein.

Last but not least: Datensicherheit

Eine wohlstrukturierte Datenstruktur ist ein Präsentierteller für Datenspionage. Zusätzlich zur strengen Abschottung der Datenbank von der Außenwelt ist die Zugriffsregelung ein wichtiger Bestandteil des Systems. Wir setzen auf datenbankinterne Zugriffssteuerung, die durch geregelte Verantwortlichkeiten für alle Datenbereiche unterstützt wird. Der Datenschutz bleibt dabei nicht außen vor und wird in jedem Fall gesondert behandelt.

Fazit

Eine Hochqualitative Datenhaltung ist für ein modernes Technologieunternehmen unerlässlich. Ein zentrales Datenbankmodell sichert bei ELMOS Semiconductor AG nicht nur die Qualität der Daten sondern auch unmittelbar die Produktqualität. Zudem hat sich im Laufe der Jahre gezeigt, dass diese Methode auch kosteneffizient ist, getreu dem Motto „Qualität lohnt sich“ [4].

Quellen

- [1] [Elmos Investor Presentation, August 2014](#) , www.elmos.com
- [2] [Zero Defects, Lockheed Martin Corporation](#), www.lockheedmartin.com
- [3] A Guide to Zero Defects: Quality and Reliability Assurance Handbook. Washington, D.C.: Office of the Assistant Secretary of Defense 1965
- [4] Quality is Free, The Art of Making Quality Certain, Philip B. Crosby 1979

Kontaktadresse:

Peter Czerner
ELMOS Semiconductor AG
Heirich-Hertz-Str. 1
D-44227 Dortmund

Telefon: +49 (0) 231 7549 160
Fax: +49 (0) 231 7549 449
E-Mail peter.czerner@elmos.com
Internet: www.elmos.com