

PEARL - Erfahrungen beim H-Bahn - Projekt der Universität Dortmund

W. Niemeyer, Siemens AG Erlangen

Einleitung

Mit Förderung des Bundesministers für Forschung und Technologie und des Ministers für Wirtschaft, Mittelstand und Verkehr des Landes Nordrhein - Westfalen ist auf dem Campus der Universität Dortmund ein H-BAHN-System errichtet worden.

1. WELCHE VERKEHRSAUFGABE IST ZU LÖSEN?

Die Universität Dortmund besteht aus zwei örtlich getrennten Bereichen, der Hauptbaufläche (HBF) und dem Aufbau- und Verfügungszentrum (AVZ), die durch eine einspurige H-Bahn-Strecke von ca. 1.050 m verkehrstechnisch verbunden sind. Die beiden Streckenenden sind über Weichen gegabelt, so daß jede der beiden Endstationen zwei Halteplätze erhält (Bild 1). Auf beiden Geländen verteilt, befinden sich die Einrichtungen wie Bibliothek, Rechenzentrum, Verwaltung, Teile der Abteilung Maschinenbau und Mensa.

Die Verkehrsbelastungen entstehen aus dem Angebot der Vorlesungen einschließlich dem Mensa-Besuch sowie durch dienstliche Fahrten von Mitarbeitern der Universität.

Die Prognose für den Verkehr zwischen den beiden Universitätsflächen ergibt eine tägliche Anzahl von durchschnittlich 4.000 Beförderungsfällen (je 2.000 Personen pro Richtungen).

2. WIE SOLL DIE VERKEHRSABWICKLUNG ERFOLGEN?

Auf der Verkehrsanlage ist ein Pendelverkehr zu realisieren, der in den Hauptverkehrszeiten mit zwei Fahrzeugen und in den Nebenverkehrszeiten mit einem Fahrzeug ausgeführt wird.

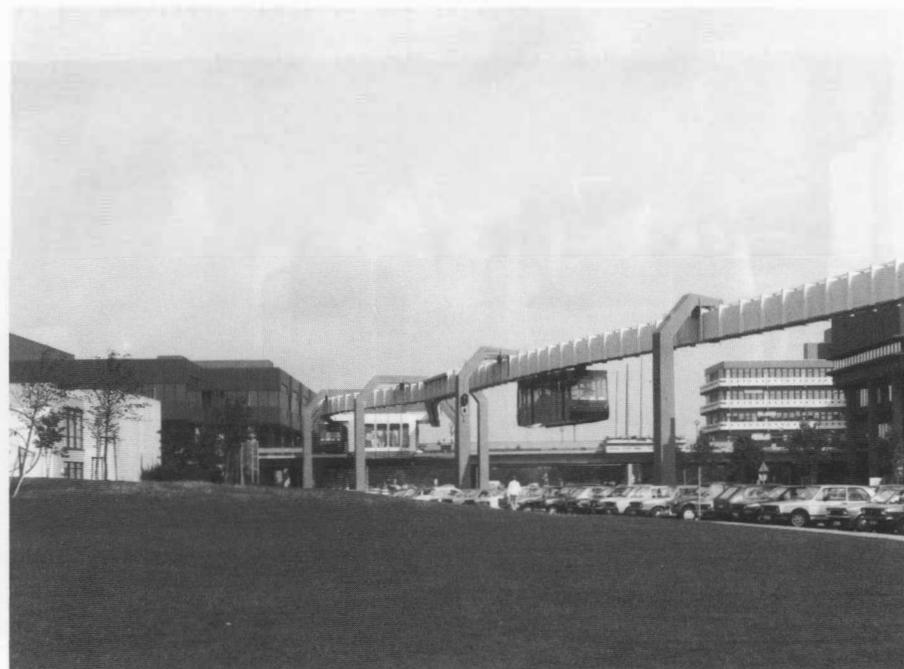


Bild 1: Stations- und Streckenbereich HBF

In Hauptverkehrszeiten soll kapazitätsorientiert in der Verkehrsart Taktbetrieb, dagegen in den Nebenverkehrszeiten bedarfsorientiert in der Verkehrsart Rufbetrieb gefahren werden (Bild 2).

Jedes Fahrzeug besitzt ein Fassungsvermögen von 20 Sitz- und 22 Stehplätzen.

Die höchste betriebliche Fahrzeuggeschwindigkeit beträgt 50 km/h.

Die Beförderungszeit von 138 s ergibt sich aus der Fahrzeit (110 s) und der Abfertigungszeit (28 s).

3. WELCHE AUFGABEN ERFOLLT DAS BETRIEBSLEITSYSTEM?

Dem Betriebsablauf der H-Bahn liegt die Verordnung über den Bau und Betrieb der Straßenbahnen (BOStrab) zugrunde, die besagt, daß die Personenbeförderung sowohl sicher als auch zügig zu gestalten ist.

Die Aufgabenbereiche gliedern sich in:

- Zentrale Disposition, Führung und Überwachung:
 - o Fahrplangestaltung
 - o Betriebsabwicklung
 - o Störungserfassung
 - o Analyse bei Betriebsunterbrechung
 - o Rettungsstrategie
- Personenbeförderung:
 - o Abfertigung des Fahrzeuges am Halteplatz sowie
 - o Abfahrt/Fahrt/Ankunft am Zielhalteplatz
- Personenschutz:
 - o Sicherung der Abfertigung und Fahrt



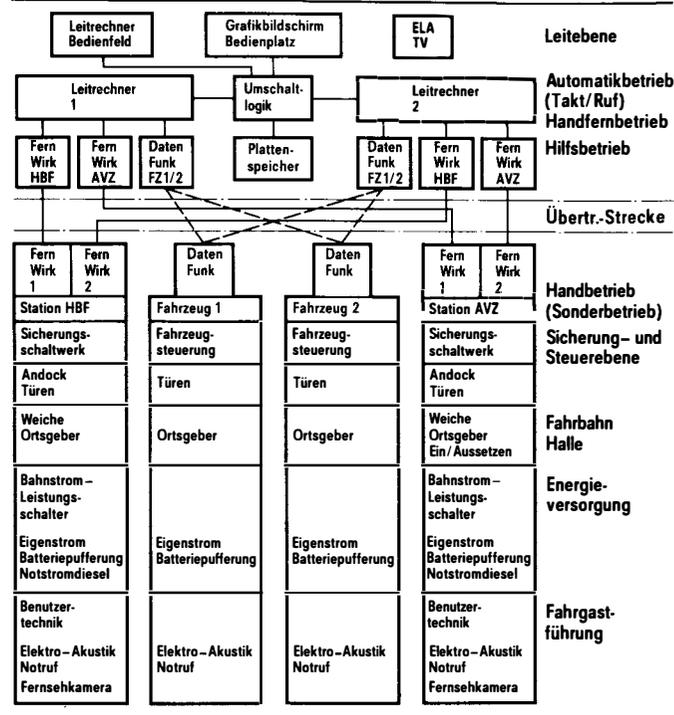
Bild 2: Benutzertechnik in der Station

4. WELCHER LÖSUNGSWEG WURDE GEWÄHLT?

Die Fahrzeuge, Stationen einschließlich Fahrstraßen werden von einem Rechnersystem überwacht und gesteuert, die sicherheitstechnisch relevanten Befehle und Kontrollen übernimmt das Sicherungssystem mit Fail-safe-Verhalten. (Bild 3: Anlagenübersichtsstruktur)

Die Leitebene ist mit einer ausgewogenen Verfügbarkeit ausgestattet, d.h. an den entscheidenden Stellen redundant ausgelegt, um

- eine hohe Verfügbarkeit zu erreichen;



**H-Bahn Dortmund – Betriebsleitsystem
Anlagenübersichtsstruktur**

EE 41
FBZ 451

Bild 3: Anlagenübersichtsstruktur

- die Reparaturausfallzeit auf ein Minimum zu reduzieren;
- die Fehlerdiagnose erheblich zu erleichtern.

5. WELCHE FUNKTIONEN ERFÜLLT DAS BETRIEBSFÜHRUNGSSYSTEM?

Das Dispositions- und Führungssystem enthält die Funk-

tionen sowohl für den vollautomatischen Betriebsablauf als auch verschiedene nicht automatische Betriebsarten, um bei Betriebsunterbrechungen von der Leitzentrale aus mittels der Störfallerfassung und -analyse dem Fahrdienstleiter ein feinstufiges Reagieren im Sinne der Rettungsstrategie zu ermöglichen.

Der Leitstand dient dem Fahrdienstleiter (Bild 4):

- zur Fahrplangestaltung, zur Information über den Betriebszustand; insbesondere zum schnellen Erkennen von betrieblichen Hemmungen sowie Störungen und zum Fernsteuern der Betriebsanlage für Inspektionsfahrten sowie zur Auflösung von Betriebsunterbrechungen über Bildschirmarbeitsplatz
- zur Kommunikation mit Fahrgästen bzw. Betriebspersonal über Sprechanlage
- zur Beobachtung des Betriebsablaufes in Stationen und auf der Fahrstraße über Fernsehanlage.

6. WELCHE VORGEHENSWEISE WIRKTE SICH BESONDERS UNTERSTÜTZEND AUF DIE SOFTWARE-ERSTELLUNG AUS?

Für die Anlagen- Betriebstechnik ist übergreifendes Systemdenken erforderlich.

In der Bearbeitungsphase AUFGABENFORMULIERUNG wurden nicht nur die Leistungsmerkmale und die Arbeitsweisen erarbeitet, sondern es erfolgt auch eine Abstimmung der Einzelfunktionen zu einem harmonischen Ganzen der einzelnen Anforderungen.

Bei dieser funktionellen Gliederung wurden von Ebene



Bild 4: Leitstand

zu Ebene (top down) auch die bekannten bzw. geforderten Eingangs-/Ausgangsdaten der jeweiligen Funktion zugeordnet und auf Vollständigkeit bzw. Widerspruchsfreiheit geprüft.

Eine darauf aufbauende Störanalyse einschließlich Berge- strategie führte rechtzeitig zur Aufdeckung von Ent-

wurfsfehlern, die in der Regel beim "Schönwetter-Denken" entstehen.

Diese Aufgabenstrukturen wurden funktionell identisch mit der Realisierungsstruktur gesetzt.

In der Praxis hat sich herausgestellt, daß die Untergliederung der Aufgabenstruktur bzw. Realisierungsstruktur des Zielsystems in vier Ebenen nicht nur technisch voll ausreicht, sondern dem Verkettungsvermögen und dem Übersichtsvermögen des jeweiligen Bearbeiters entgegenkommen.

Die genannten vier Ebenen der Aufgaben- und Realisierungsstruktur des Projektstrukturplanes enthalten:

	Aufgabenstruktur	Realisierungsstruktur
Ebene 1	Aufgabe	System
Ebene 2	Teilaufgabe	Subsystem
Ebene 3	Funktion	TASK (Modul)
Ebene 4	Unterfunktion	Baustein (Prozedur)

Für die Spezifikation wurden rechnergestützte Werkzeuge verwendet. Über sie wurde die erforderliche tabellarische und grafische Dokumentation schritt haltend erstellt.

Zur Erleichterung für die Erstellung einer einheitlich prüf- baren Dokumentation wurden bildschirmgerechte Vor- drucke benutzt.

7. WELCHE SOFTWARE-WERKZEUGE UND TESTHILFEN KAMEN ZUM EINSATZ?

Die Voraussetzung für den Einsatz von Software-Werk- zeugen und Testhilfen ist durch die Ausstattung der

H-Bahn-Doppelrechneranlage mit Zusatz-Peripherie (Schnelldrucker, Plotter) geschaffen worden.

Zu den nützlichen und effektiven Werkzeugen und Hilfen zählen nachstehende Funktionen:

(siehe Bild 5: Software-Werkzeuge und -Testhilfen)

- Editor MEDIS (Multi-User EDitor System) wird zur Textverarbeitung für die Software-Erstellung und Dokumentation

- Informationsverarbeitung
- Inbetriebsetzungs-Strategie
- Terminpläne
- etc.

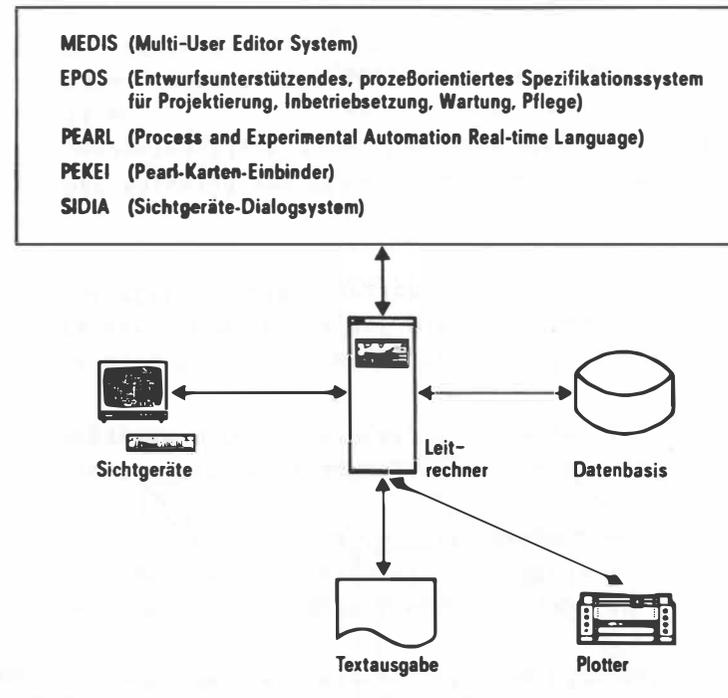
verwendet.

- EPOS (entwurfsunterstützende, prozeßorientiertes Spezifikationssystem für Projektierung, Inbetriebsetzung, Wartung, Pflege)

Zur Entwurfsobjekt-Beschreibung (grob- bis feinstufig) mit Analyse und Prüfung sowie Generierung der Dokumentation mit dem Ergebnis schritthaltender grafischer Darstellungen (Blockdiagramme, Datenflußpläne) über Plotterausgabe sowie zugehörige Referenzlisten über Schnelldruckerausgabe.

- PEARL-Compiler (Process and Experimental Automation Real time Language) in Verbindung mit dem Änderungshilfsmittel PEKEI (PEARL-Karten-Einbinder) zur Änderung von Deklarationen in Spezifikationen für alle in Verbindung stehenden Monaten.

- SIDIA Sichtgeräte-Dialogsystem für Test- und Hilfsfunktionen, wie



Software-Werkzeuge und -Testhilfen

- Prozeßbedienung, Fahrplangestaltung
- Prozeßparametrierung
- Ablaufverfolgung
- Prozeßabbildprotokollierung
- Modultest/Simulation

über benutzerfreundliche Bedienung mittels Masken am Sichtgerät.

8. WARUM HABEN WIR PEARL ANGEWENDET?

PEARL erschien uns aus folgenden Gründen vorteilhaft:

- Mit Vorkenntnissen in höheren Programmiersprachen ist PEARL 300 leicht erlernbar. In der allgemeinen Literatur ist ein zunehmender Verbreitungsgrad erkennbar. Ein ausgereifter Standard liegt vor und ist durch spezielle Siemens-Beschreibungen dokumentiert.
- Es besitzt die für uns erforderlichen Eigenschaften wie Echtzeitsprachmittel, problemangepaßte Datentypen insbesondere Bit-Daten, Signals zum Abfangen von Laufzeitbesonderheiten etc.
- Auf Anwendererfahrungen mit PEARL kann zurückgegriffen werden.
- Es gibt umfangreiche Systemunterstützung durch Standardprogramme wie:
Betriebssystem ORG 300,
Editor MEDIS 300
PEARL COMPILER PC 30
Binder BD 30 bzw. LNKR
Datei- und Bibliotheksverwaltung FILE.

9. WELCHE VORAUSSETZUNGEN UND BEDINGUNGEN GAB ES FÜR DEN PEARL-EINSATZ?

Als Basis für das Anwenderssoftwaresystem wurde der Softbus als Koordinator (KOOR) in Verbindung mit dem Sichtgerätedialogsystem SIDIA eingesetzt.

Das Ziel des im Softbus enthaltenen Konstruktionsprinzips ist es die nahezu unendliche Freiheit, eine Soft-

ware zu entwerfen, in eine erprobte Bahn zu lenken, welche dem Entwickler trotzdem den notwendigen Spielraum läßt. Die Grundgedanken sind:

- o Modulare Softwaresysteme prinzipiell so aufzubauen wie Hardware-Baugruppensysteme:

Einbaurahmen	= Rechner
Bus (Verdrahtung)	= Softbus (Koodinator) (s. Bild 6)
Baugruppe	= Modul (TASK)

was bedeutet, daß Moduln untereinander nur über den Softbus verkehren und dadurch alle eine einzige, logisch gleiche, interne Schnittstelle haben.

Auf dem Softbus basiert das Dialogsystem SIDIA zur Erstellung von Betriebsdialogen mittels virtueller Tasten sowie Protokollen auf Grafiksichtgeräten.

Die Anwendung des Softbuses (KOOR) erlaubt die technologische Aufgabe in PEARL zu programmieren.

Um eine Kollisionsgefahr zwischen Softbus und PEARL zu vermeiden, wurde auf die Koordinationseigenschaften von PEARL-TASKING verzichtet.

Der Anschluß von PEARL-Programmen an den Softbus ist unproblematisch durch Einbinden des CALL-Aufrufes möglich.

Alle mit PEARL 300 gewonnenen Erfahrungen basieren auf der PEARL-Variante "D".

10. WELCHE ERFAHRUNGEN HABEN WIR MIT PEARL GEMACHT?

Die bisher mit PEARL gewonnenen Erfahrungen sind überwiegend positiv zu bewerten. Im einzelnen seien hierfür folgende Erkenntnisse genannt:

- Einfaches Compiler-Handling.
- Kurze Kompilierungszeiten.
- Ausgabe der Syntaxfehler im Klartext mit Fehlerlokalisierung.
- Selbstdokumentierendes Compiler-Listing bei strukturierter Programmierung.
- Modularität auf Quellsprache-Ebene; damit werden Neuübersetzungen auf das entsprechend überarbeitete PEARL-Modul beschränkt, was in der Praxis zu erheblichen Einsparungen von Rechenzeit führt.
- Problemloser Assembleranschluß.
- Optimale Anpassungen von PEARL an die Systemumgebung der Prozeßrechner PR 300. Dabei stützt sich der PEARL-Compiler voll auf die Funktionen der PR 300-Betriebssysteme ab.
- Geringe Fehlerhäufigkeit infolge der PEARL-Blockstruktur und Mächtigkeit der PEARL-Sprachelemente.
- Selektives Einbinden des PEARL-Laufzeitsystems (linear) ermöglicht es, daß nur die Laufzeitroutinen in das entsprechende PEARL-Anwenderprogramm eingebunden werden, die das Programm benötigt.
- Reduzierung des Adressraumes bei nicht-linearem Hinzubinden der PEARL-Library:
Das Laufzeitsystem von PEARL besteht zum größten Teil (ca. 16,5 KW aus Ablaufinvarianten-Moduln (8 Großmoduln von PEARL)). Wenn das Lauf-

zeitsystem jedes PEARL-Programm linear zugebunden würde, so müßte n-mal der entsprechende Hauptspeicherplatz in Anspruch genommen werden (sehr speicherintensiv).

Deshalb werden die 8 Großmoduln von PEARL vorgebunden als ablaufinvarianter Common-Code (CC) in den Common-Bereich (CB) ausgelagert. Das Einbinden erfolgt dann über die jeweiligen Übersetzungstafeln der Programme bzw. Programmpakete beim Laden. Der Adressraum wird durch

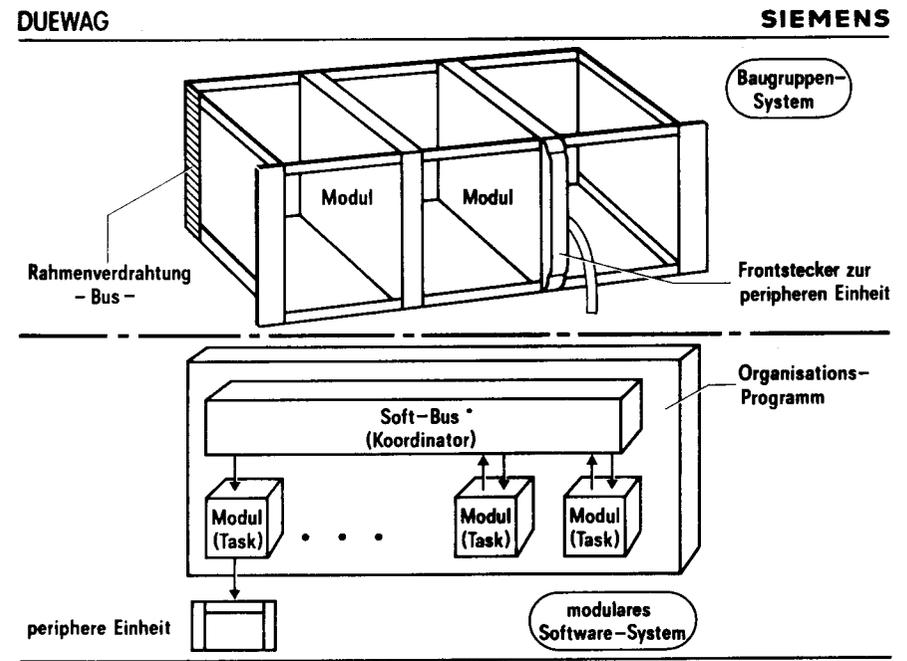


Bild 6: Analogie: Baugruppensystem - modulares Softwaresystem

das Laufzeitsystem für PEARL-Objekte von maximal 64 KW auf ca. 47 KW reduziert. Unter diesen Voraussetzungen ist je PEARL-Programm mit einem Overhead von 1 bis 2 KW für Gerätetreiber zu rechnen.

- Der vom Compiler abgesetzte Code schwankt in seinem relativen Speicherbedarf sehr stark in Abhängigkeit des jeweils individuellen Programmierstils. Im Vergleich mit ASSEMBLER liegt der Faktor 1:3 bis 1:4 sicher im Rahmen der Realität.

Was wäre aus unserer Sicht verbesserungswürdig?

- Anpassung von Datentypen:
Aus Sicherheitsgründen können nur Daten gleichen Typs miteinander verknüpft werden. Soll dies mit datenungleichen Typs erfolgen, so ist eine Längen- und Typanpassung notwendig. Eine echte Konvertierung in den jeweiligen Zieldatentyp ist nur zwischen FIXED und FLOAT gegeben. Ferner muß bei einigen Typanpassungen ein Umweg über völlig unbeteiligte Datentypen gemacht werden. Eine echte Konvertierung zwischen Charakter, FIXED und FLOAT, wäre wünschenswert (z.B. TOFIXED '4711' CHAR (4)).
- Reaktion des Anwenders auf Standardsignale von PEARL:
Für Programmlaufbesonderheiten stellt PEARL eine Reihe von Standardsignalen zur Verfügung. Der Anwender hat die Möglichkeit mit der 'ON-Anweisung' auf diese entsprechend zu reagieren. Da der Standardmeldung des Laufzeitsystems nicht

eindeutig entnommen werden kann, wo und wann das Standardsignale auftrat, bleibt dem Anwender oft keine andere Wahl, als diese im Laufe der Testphase empirisch zu ermitteln (z.B. Programmnummer PEARL SIGNAL WO:5). Eine deutliche Verbesserung wäre durch zusätzliche Angabe der virtuellen Adresse des Feldschreibungsblocks zu erreichen.

- Länge von Bit-Variablen:
Die Länge von Bit-Variablen ist auf maximal 16 Bit begrenzt. Im Hinblick auf ein besseres Daten-Handling wäre eine Erweiterung wünschenswert.
- Testsystem:
Für den Einzeltest von PEARL-Programmen steht ein Minitest-System zur Verfügung. Die Testentschlüsse können vom Compiler auf Wunsch des Anwenders in das Testobjekt eingebracht werden. Das Testobjekt kann selektiv oder auf jeder PEARL-Zeile angehalten und fortgesetzt werden. Ferner kann eine Ablaufverfolgung (TRACE) mit Protokollierung auf Sichtgerät vorgenommen werden. Für Datenmanipulationen muß die übliche Testhilfe (TEST 300 (S)) benutzt werden. Die virtuellen Adressen von globalen PEARL-Daten sind dem Binderprotokoll zu entnehmen. Für umfangreiche Systemtests unter Online-Bedingungen steht neuerdings das leistungsstärkere System, DEBUG, zur Verfügung.

11. ZUSAMMENFASSENDE BEWERTUNG

Der Einsatz von PEARL im H-Bahn-Projekt Dortmund hat sich positiv auf die Qualität des Anwender-Softwaresystems ausgewirkt. Die weitere Anwendung von PEARL bei Folgeprojekten im Zusammenhang mit den erwähnten Software-Werkzeugen ist nicht nur empfehlenswert, sondern eine notwendige Vorgehensweise zur Effektivitätssteigerung.