

**Warum  
sind quelloffene,  
betriebswirtschaftliche  
Programme Mangelware?**

**EDV-Lotse Dimitri Asarowski  
5.11.2017**

## Inhalt

1. Fakten
2. Meinung aus der Community
3. Kundenerwartungen
4. Rechtliche Aspekte
5. Wirtschaftliche Aspekte
6. Technologische Aspekte

## Fakten

### Betriebssysteme

Microsoft Windows, Apple Mac OS X

BSD Unix, Linux

### Virtualisierung

VMWare, Microsoft Hypervisor

Xen, KVM, Oracle Virtualbox

### Datenbanken

Oracle, IBM DB2, MS SQL, Sybase

MySQL/MariaDB, PostgreSQL, FireBird

### Browser

Internet Explorer, Safari, Opera, Chrome

Mozilla Firefox, Chromium

### Betriebswirtschaftliche bzw. Branchenlösungen

> 1000

~ 40

## Meinung aus der Community

Betriebswirtschaftliche

Programme

sind

nicht

**SEXY**

## **Kundenerwartungen**

Kunden fragen nicht nach Open-Source  
sondern suchen ein Allheilmittel.

- A. Einsatzfähigkeit
- B. Bekanntheitsgrad
- C. Nachhaltigkeit
- D. Bezahlbarkeit
- E. Flexibilität

## System Stack



Anwendungssoftware

Compiler / Interpreter / Framework

FIRMWARE / Virtualisierungssoftware

HARDWARE

## Rechtliche Aspekte

	Proprietär	Proprietär + Quellcode	Open Source
Lesen	-	+	+
Ändern	-	+	+
Verbreiten	-	-	+

## Wirtschaftliche Aspekte

	Proprietär	Open Source
Entwicklung	- XX XXX	-X XXX
Verkauf (Lizenzen)	+ XXX XXX	0
Implementierung	+ XXX	+X XXX
Wartung und Support	+ XX	+XXX

## Anforderungen an Entwicklungswerkzeuge

- A. Komplexe Datenstrukturen  
und Geschäftslogik
- B. Datenspeicherung und -verarbeitung
- C. Schnittstellen und Interoperabilität
- D. Visuelle Programmierung  
und Modellierung

## **COBOL (1959)**

**CO**mmun **B**usiness **O**riented **L**anguage

1. ADD x TO y GIVING z

2. Datenstrukturen

3. Dezimale Arithmetik, Formatierung

4. Erweiterte Eingabe/Ausgabe

## Objektorientierte Programmierung

Simula (1967)

Smalltalk (1969-1980)

Objekt = Daten + Methoden

Object Rechnung

Nummer, Datum, Kunde, Warenliste

funktion Erstellen

funktion Buchen

funktion Stornieren

## **Tabellenkalkulation**

VisiCalc (1979) auf Apple II

Lotus 1-2-3 (1983) auf IBM PC

Microsoft Excel (1987) für Windows 2.0

**Endbenutzer-orientiert**

## **xBase**

dBASE II (1980) - dBASE III (1986)– dBASE IV

FoxBASE (1984) – FoxPRO (1989-2007)

Clipper (1985 – 1997) - Compiler

Harbour (1999) – Open Source Compiler

1. DBF- offene Dateiformat für Filedatenbank
2. Entwicklungs- und Ausführungssystem
3. Mehrbenutzerfähige Anwendungen

## SQL-Datenbanken

**Structured Query Language** ist eine deklarative Sprache für Datenverwaltung in relationalen Datenbanken

Oracle (1979)

PL/SQL (1992) Oracle7

OSS: MySQL/MariaDB, PostgreSQL, Firebird

## **ABAP/4 (1983)**

**Allgemeiner Berichtsaufbereitungsprozessor**  
**Advanced Business Application Programming**

1. Integration von Datenbank- und Benutzerschnittstellenanweisungen in die Programmiersprache
2. Interaktive Entwicklungsumgebung (ABAP-Workbench)
3. Plattform für Geschäftsanwendungen (NetWeaver)

## Lotus Notes (1989)

ist ein dokumentenorientiertes Datenbanksystem

erstmal eine Message-Board-Anwendung

dann Formelsprache und danach LotusScript

Plattformunabhängig (Windows, Mac OS X, Linux)

Skalierbar

Verteilte Datenbanken

## ObjectVision (1991)

ist eine visuelle Entwicklungsumgebung.

Eine ObjectVision-Anwendung besteht aus grafisch gestalteten Formularen, die Objekte und Ereignisse enthalten, um Interaktivität bereitzustellen. ObjectVision-Anwendungen können auch mit Datenbanken interagieren, wie Paradox und dBase. Ein fertiges Projekt wird als OVD-Datei gespeichert, die von einer interpretierten Laufzeit ausgeführt wird und frei verteilt werden kann.

ObjectVision wurde außer in einigen Nischensegmenten nicht breit verwendet. Die visuellen Programmierideen waren aber die Grundlage für Borland Delphi.

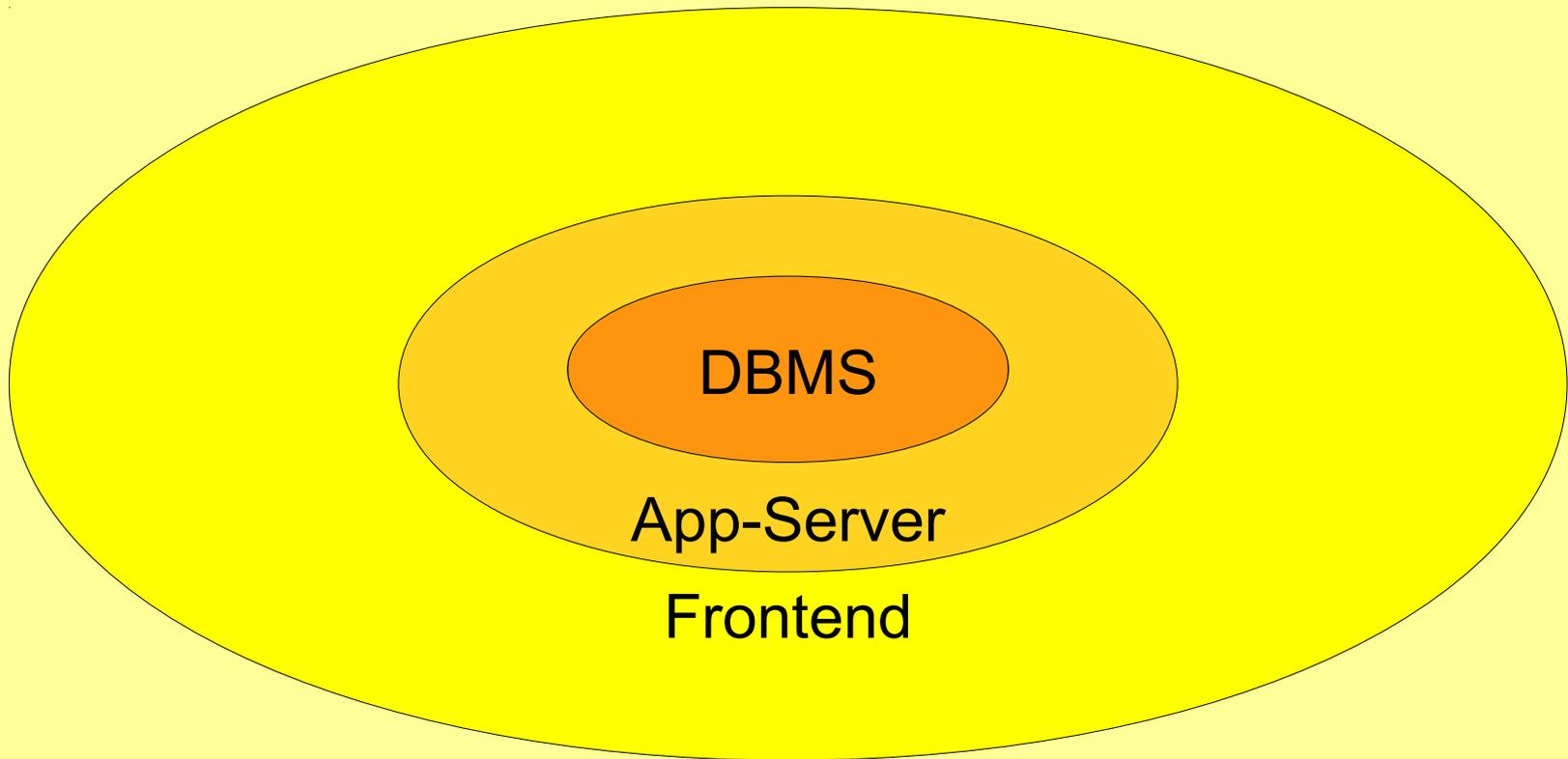
## Rapid Application Development (RAD)

ist eine schnelle Anwendungsentwicklung. Das Konzept wurde in den 1980er Jahren von Barry Boehm zur Softwareentwicklung mit einem prototypischen Vorgehensmodell miterdacht. Es dient dazu, Softwareentwicklung im Vergleich zu klassischen Vorgehensmodellen wie dem Wasserfallmodell flexibler zu gestalten und sich an schnell ändernde Anforderungen anpassen zu können.

Microsoft Visual Basic (1991)

**Borland Delphi (1995)**

**XAMPP**



## Berufsbild (Wirtschafts) Informatiker/in



**Christoph**  
Informatiker

Wir haben das Framework nun fertiggestellt und koennen das Upgrade der SQL-Engine slipstreamen!



**Michael**  
Wirtschaftsinformatiker

Ich uebersetze das mal...

**Einsatzgebiete von (Wirtschafts)Informatikern sind auesserst vielfaeltig!**

**Die Geschichte der (Wirtschafts)Informatiker ist jung, aber lang!**

**Auch die Anforderungen und Faehigkeiten in diesen Berufen sind sehr unterschiedlich und interessant!**

Die Informatik spricht ihre ganz eigene Sprache. Damit sowohl der Finanzvorstand, als auch Frau Meier aus dem Buero nebenan weiss, was Christoph genau mit seiner Aussage meint, uebersetzt Michael immer gerne...

Boris, Abderrahim, Martin, Dominik, Francesco, Julian, Ali, Tobias, Evgenij, Dennis, Irina, Amelie, Marcel, Stefka, Thomas, Mika



## **Keine „Silberkugeln“**

*"Es gibt keine einzige Entwicklung, weder technologisch noch bei den Techniken des Managements, die für sich allein versprechen könnte, im Laufe von zehn Jahren wenigstens eine Größenordnung [das Zehnfache] an Verbesserungen bei Produktivität, Zuverlässigkeit oder Einfachheit zu garantieren."*

***Frederick P. Brooks (1986)***

**Wesentlicher (Essence) Komplexität**

**Zufälliger (Accident) Komplexität**

## Domain-Driven Design (DDD)

ist eine Herangehensweise an die Modellierung komplexer Software. Die Modellierung der Software wird dabei maßgeblich von den umzusetzenden Fachlichkeiten der Anwendungsdomäne beeinflusst. Der Begriff „Domain-driven Design“ wurde von Eric Evans in seinem Buch (2003) geprägt.

Entitäten (Entities, reference objects)

Wertobjekte (value objects)

Aggregate (aggregates)

Serviceobjekte (services)

Module (modules, packages)

Ubiquitäre Sprache (ubiquitous language)

## Domain-Specific Language (DSL)

Anwendungsspezifische Sprache ist eine formale Sprache, die zur Interaktion zwischen Menschen und digital arbeitenden Computern („Computersprache“) für ein bestimmtes Problemfeld (die sogenannte Domäne) entworfen und implementiert wird. Beim Entwurf einer DSL wird man bemüht sein, einen hohen Grad an Problemspezifität zu erreichen: die Sprache soll alle Probleme der Domäne darstellen können und nichts darstellen können, was außerhalb der Domäne liegt. Dadurch ist sie durch Branchenexperten ohne besonderes Zusatzwissen bedienbar.

ABAP

TROIA

1C BSL

SAP NetWeaver

Applikationsserver

1C:Enterprise

SAP ERP

caniasERP

diverse + DIY



Dimitri Asarowski

Mendener Str. 79

53757 Sankt Augustin

Telefon: 02241 - 932 26 36

Telefax: 02241 - 80 69 19

E-Mail: [info@edv-lotse.de](mailto:info@edv-lotse.de)