

# Messen mit Strobe®

## Workshop / Kurzeinführung

**cps4it**

consulting, projektmanagement und seminare für die informationstechnologie

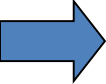
Ralf Seidler, Stromberger Straße 36A, 55411 Bingen

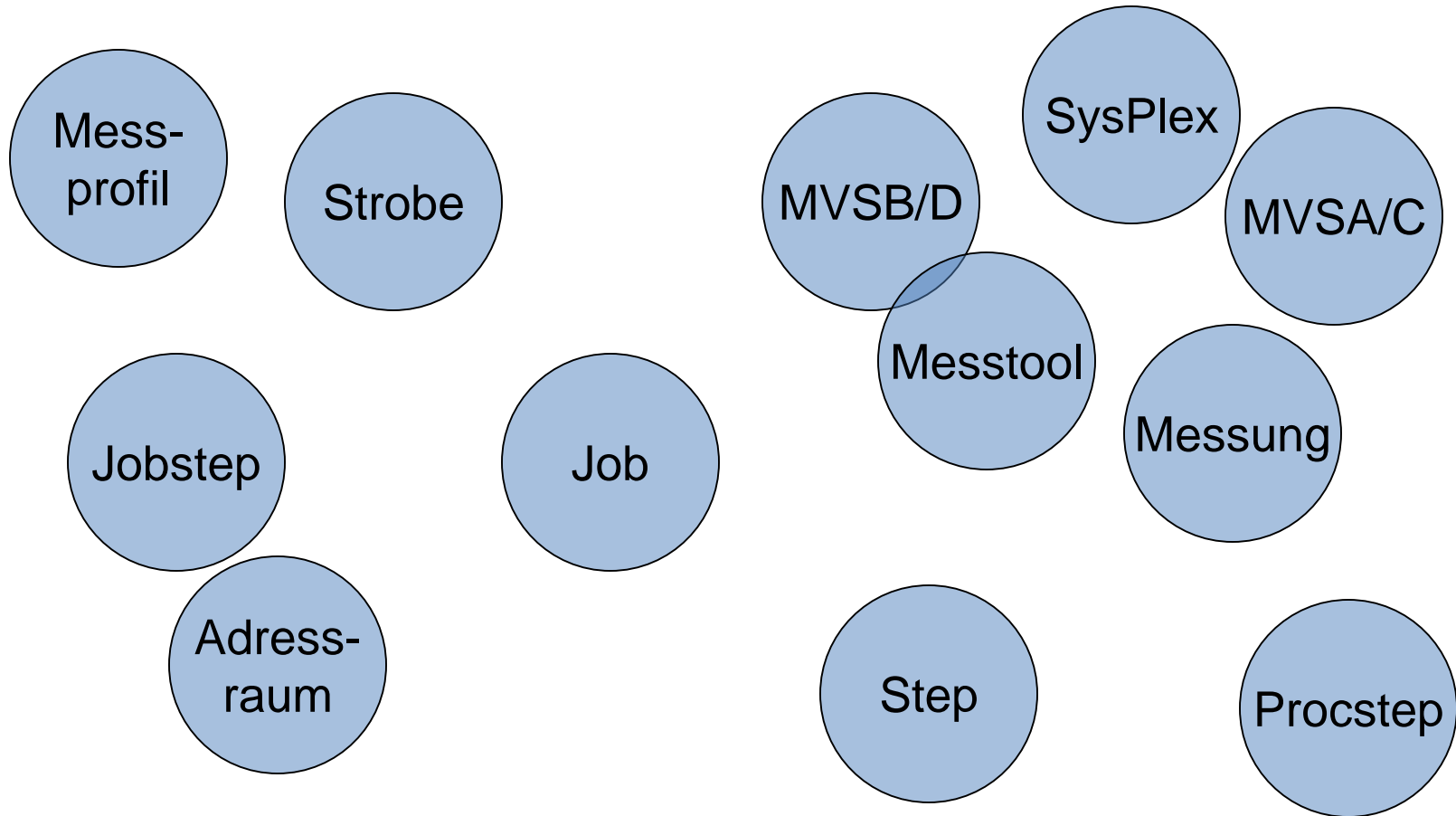
Fon: +49-6721-992611, Fax: +49-6721-992613, Mail: [RSeidler@cps4it.de](mailto:RSeidler@cps4it.de)

Internet: <http://www.cps4it.de>

- 
- Grundbegriffe eines Messtools und seinen Umgang kennen lernen
    - einfache Messungen analysieren
  - Möglichkeiten, performante Anwendungen in COBOL zu erstellen und zu warten, kennen lernen (nur Literaturhinweise)
    - COBOL-Code
    - Compile-Optionen und LE-Optionen
    - DB2-SQLs und DB2-Umgebung

- 
- Seite 5: Vorstellung und Einführung
  - Seite 13: Optimierungen – Beispiele / Potential
  - Seite 27: Strobe – Handling
  - Seite 61: Vorgehensweise Analyse (Strobe)
  - Seite 95: Vorgehensweise Analyse (iStrobe)
  - Seite 113: Modellierung und DB2-Zugriffe
  - Seite 135: COBOL–Felder – COBOL-Befehle
  - Seite 143: Optionen – COBOL – LE
  - Seite 148: Beispiele – Diskussion – Austausch

- 
- A blue arrow pointing to the right, highlighting the first item in the list.
- Vorstellung und Einführung
  - Optimierungen – Beispiele und Potential
  - Strobe – Handling
  - Vorgehensweise für die Analyse (Strobe)
  - Vorgehensweise für die Analyse (iStrobe)
  - Modellierung und DB2-Zugriffe
  - COBOL–Felder – COBOL-Befehle
  - Auswirkungen von Optionen – COBOL – LE
  - Praxisbeispiele – Diskussion – Austausch



- PMA
  - Performance Management für Anwendungen
- PROP
  - Performance- und Ressourcenorientierte Programmierung
- APM
  - Application Performance Management

- Strobe®
  - messen
  - Profil erzeugen
  - Oberfläche ISPF / Eclipse Plug-In
- Autostrobe® – Zusatzprodukt
  - automatisieren von Messungen
- iStrobe®
  - Interpretation der Profile
  - Oberfläche Browser

- **COBOL-Optimierungen – Potential allgemein**
  - Perf.Paper von IBM zu V4.2:  
<http://www.ibm.com/support/docview.wss?rs=203&q=7018287&uid=swg27018287>
- **COBOL Compile Options**
  - Enterprise COBOL for z/OS Programming Guide Version 4.2 Kapitel 17  
<http://publibfp.boulder.ibm.com/epubs/pdf/igy3pg50.pdf>
- **LE Options**
  - z/OS Language Environment Programming Reference: Kapitel 2  
<http://publibz.boulder.ibm.com/epubs/pdf/cee13200.pdf>
- **COBOL–Code**
  - Enterprise COBOL for z/OS Programming Guide Version 4.2 Kapitel 34  
<http://publibfp.boulder.ibm.com/epubs/pdf/igy3pg50.pdf>
  - Perf.Paper von IBM zu V4.2:  
<http://www.ibm.com/support/docview.wss?rs=203&q=7018287&uid=swg27018287>
- **DB2**
  - DB2 10 for z/OS Managing Performance (SC19-2978-08)  
<http://publib.boulder.ibm.com/epubs/pdf/dsnpgm08.pdf>
  - DB2 Version 9.1 for z/OS Monitoring and tuning DB2 performance  
[http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp?topic=/com.ibm.db29.doc.perf/db2z\\_perf.htm](http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp?topic=/com.ibm.db29.doc.perf/db2z_perf.htm) - gibt es schon nicht mehr ☹

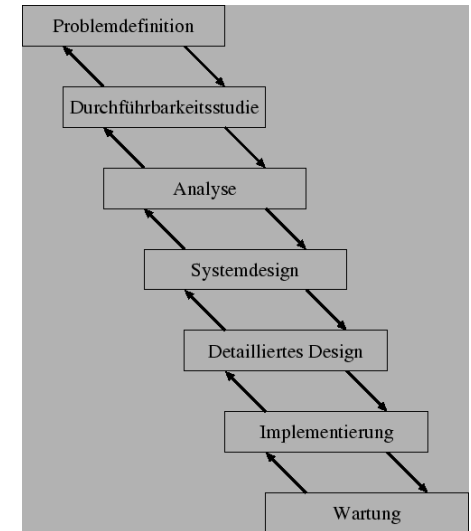


- Strobe – Handling
  - <http://frontline.compuware.com/>  
nur für registrierte Benutzer
    - STROBE MVS User's Guide
    - STROBE for CICS / for DB2 / for IMS
  - Bookmanager im Hause
- Strobe – Interpretation
  - !! Es gibt keine Literatur vom Hersteller, wie Strobe-Profile interpretiert werden !!
- PMA / PRoP bei der RRR
  - vielleicht bald im Intranet ;-)
- <http://wiki.cps4it.de>



## Lifecycle der Anwendungen (Wasserfallmodell / V-Modell)

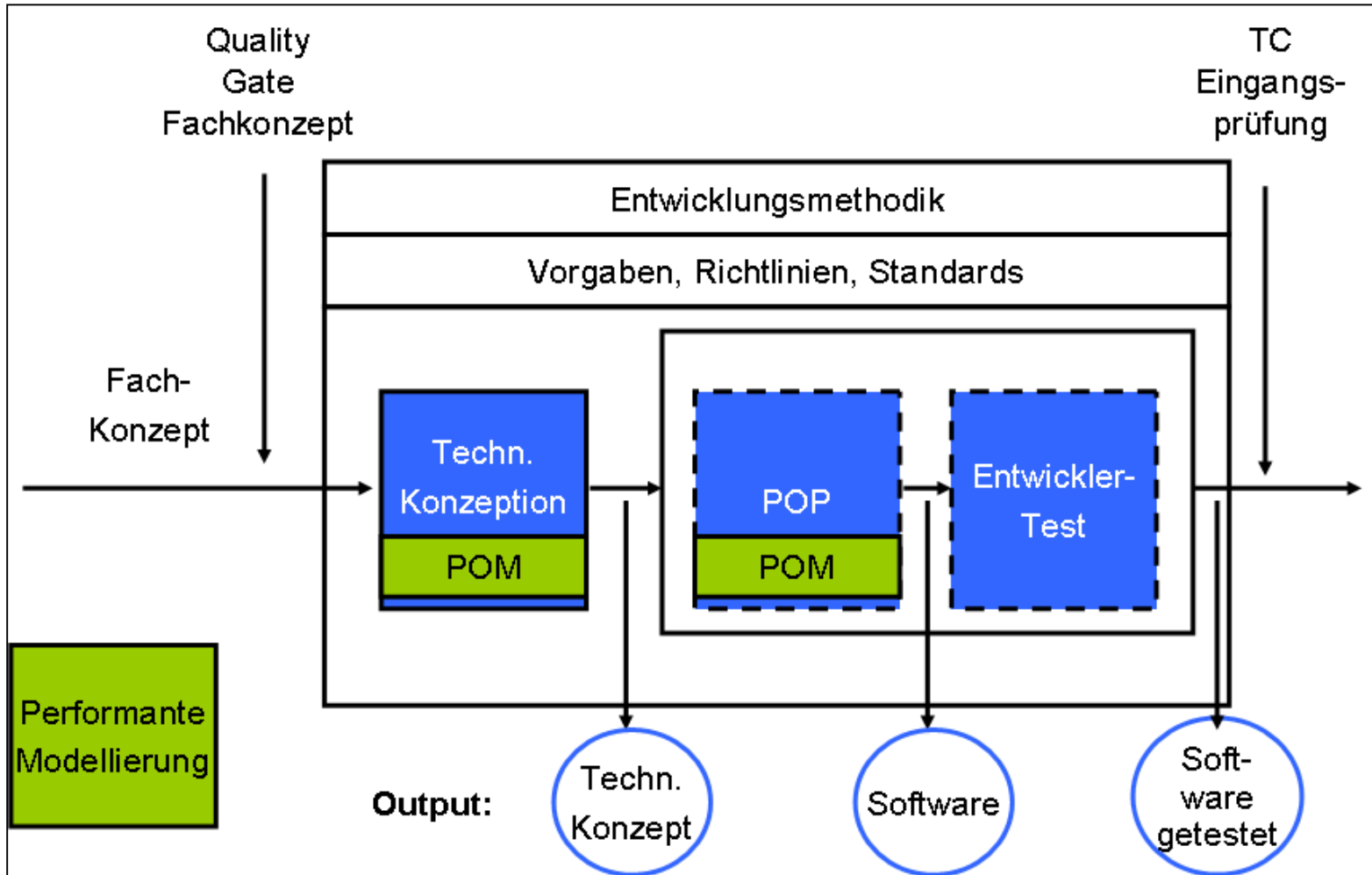
- Planung / Grobentwurf
- Analyse / Fachentwurf
- Design / technischer Entwurf
- Programmierung mit Modultest
- Integration und Systemtest
- Auslieferung, Einsatz und Wartung



**PROP / PMA / APM**



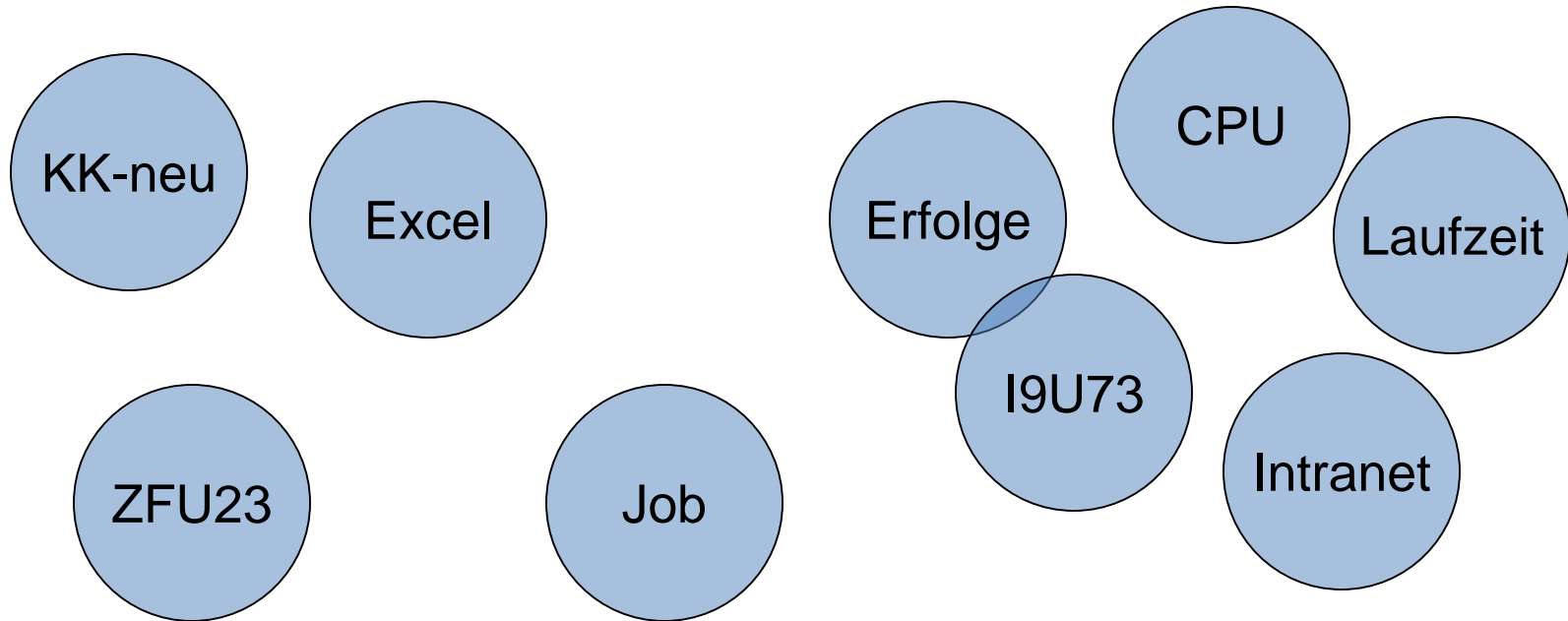
## RRR-Schaubild



- 
- Vorstellung und Einführung
  - ➔ • Optimierungen – Beispiele und Potential
  - Strobe – Handling
  - Vorgehensweise für die Analyse (Strobe)
  - Vorgehensweise für die Analyse (iStrobe)
  - Modellierung und DB2-Zugriffe
  - COBOL–Felder – COBOL-Befehle
  - Auswirkungen von Optionen – COBOL – LE
  - Praxisbeispiele – Diskussion – Austausch

## Begriffe

---

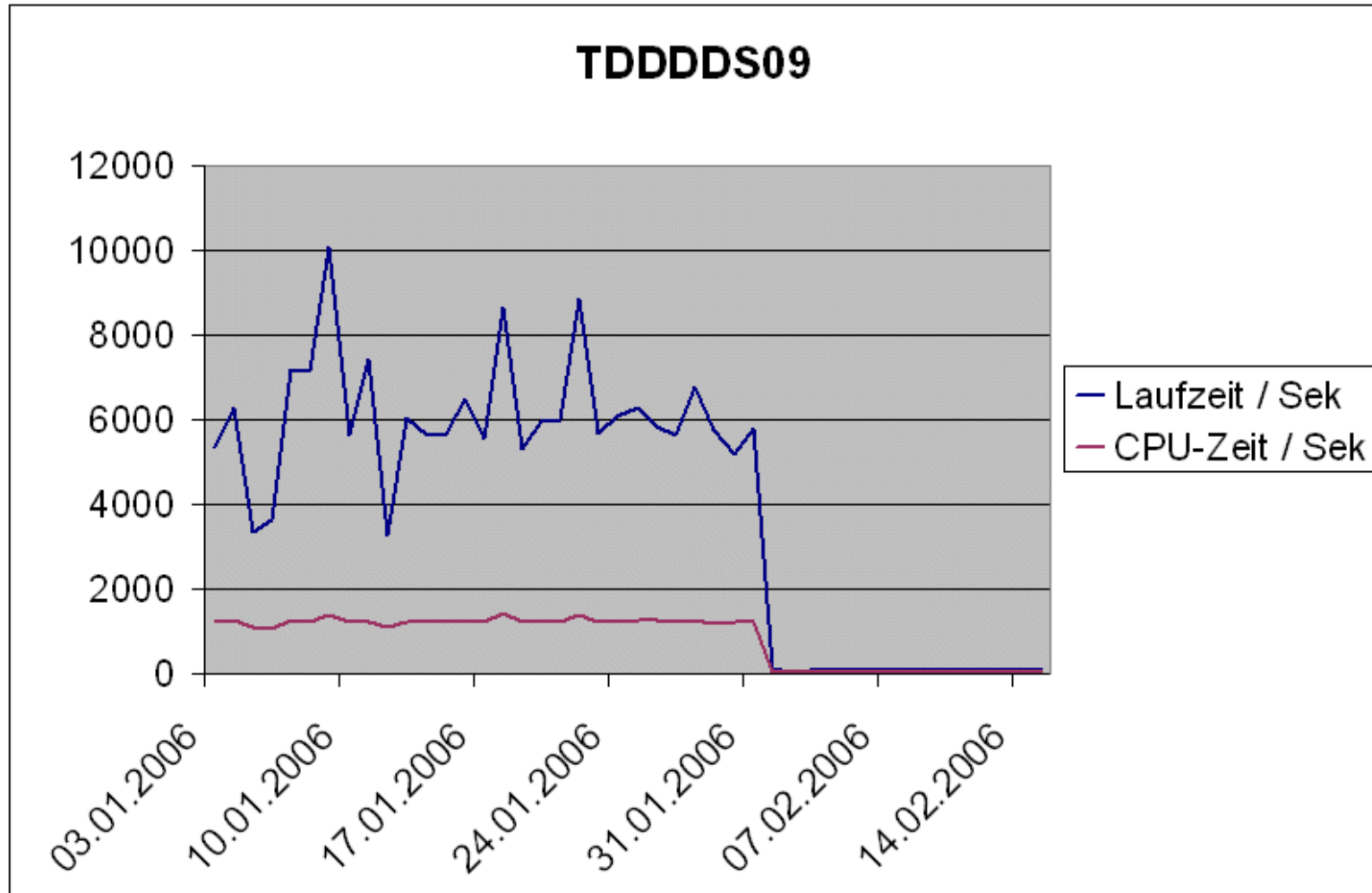


# Optimierungen – Beispiele und Potential

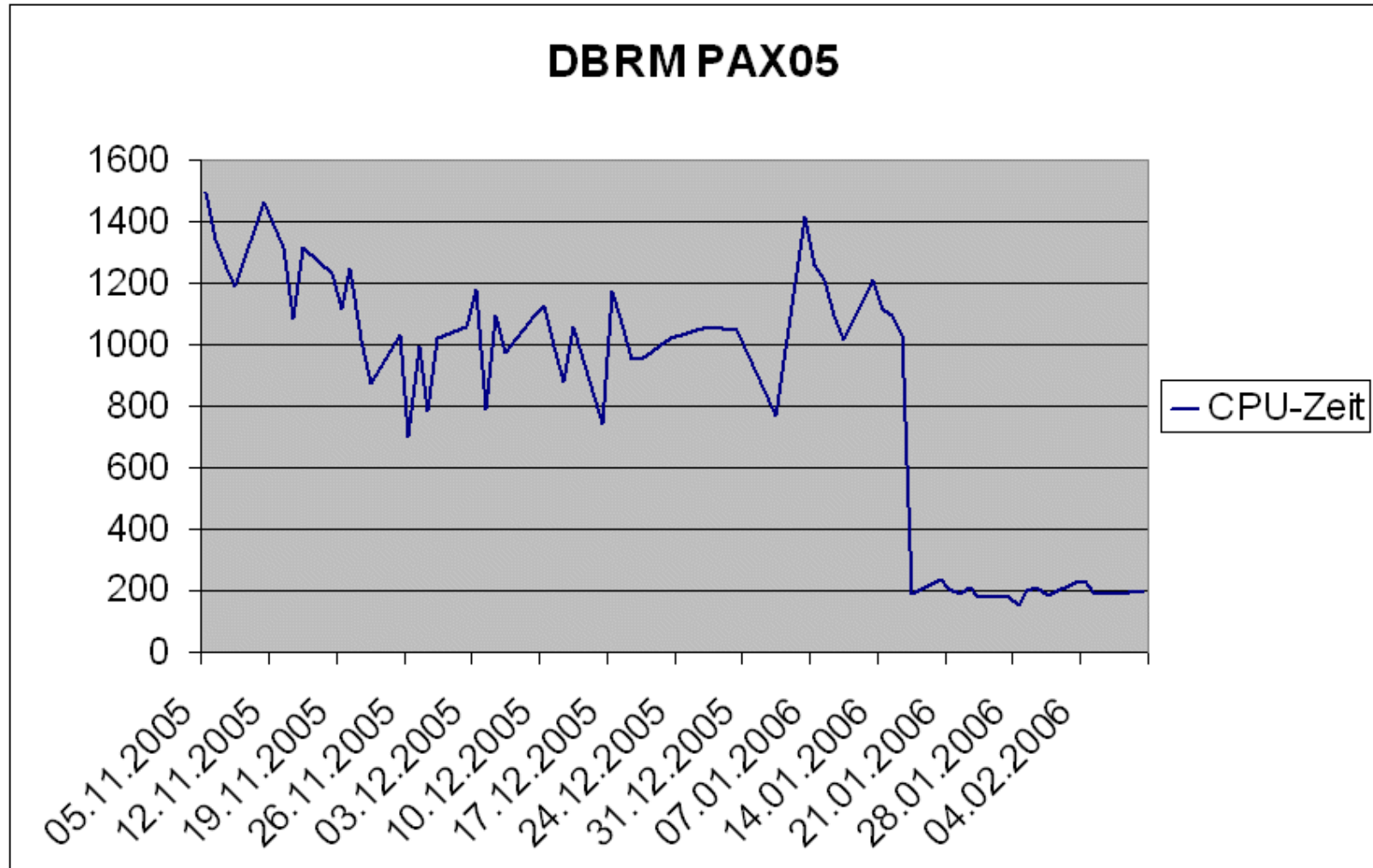
## Beispiel 1 – Aufrufhäufigkeit zentrales Modul

Programm / Job	CPU Ersparnis (hochgerechnet auf 1 Jahr)	Laufzeit Ersparnis (hochgerechnet auf 1 Jahr)	Art der Optimierung
I9L55	1000 Stunden	(Online)	versch.
GTE71	300 Stunden	(Online)	Index eingeführt
DO692/PDODO692	700 Stunden	850 Stunden	Reorganisation DB2-Table
I9S61 / IMS	240 Stunden	(Online)	Loop (Code).
<b>N1451 / TN1451*</b>	<b>270 Stunden</b>	<b>336 Stunden</b>	<b>Aufrufhäufigkeit N2U73</b>
GT500 / MGT500%1	--- <sup>(3)</sup>	--- <sup>(3)</sup>	RUNSTATS, REBIND
BF001 / MBFBF001	240 Stunden	310 Stunden	DB2-Optimierung
<b>N2735 / TN2735*</b>	<b>625 Stunden</b>	<b>667 Stunden</b>	<b>Aufrufhäufigkeit I9U73</b>
Posy / TPOPO001	- - -	30 Stunden	BUFNO=16
<b>IK019 / TIKIK019</b>	<b>40 Stunden</b>	<b>120 Stunden</b>	<b>Aufrufhäufigkeit I9U73</b>
	<b>3.317.559,12 €<sup>(1)</sup></b>		

## Beispiel 2 – SQL-Änderung Tagesjob



## Beispiel 3 – SQL-Änderung zentrales Modul



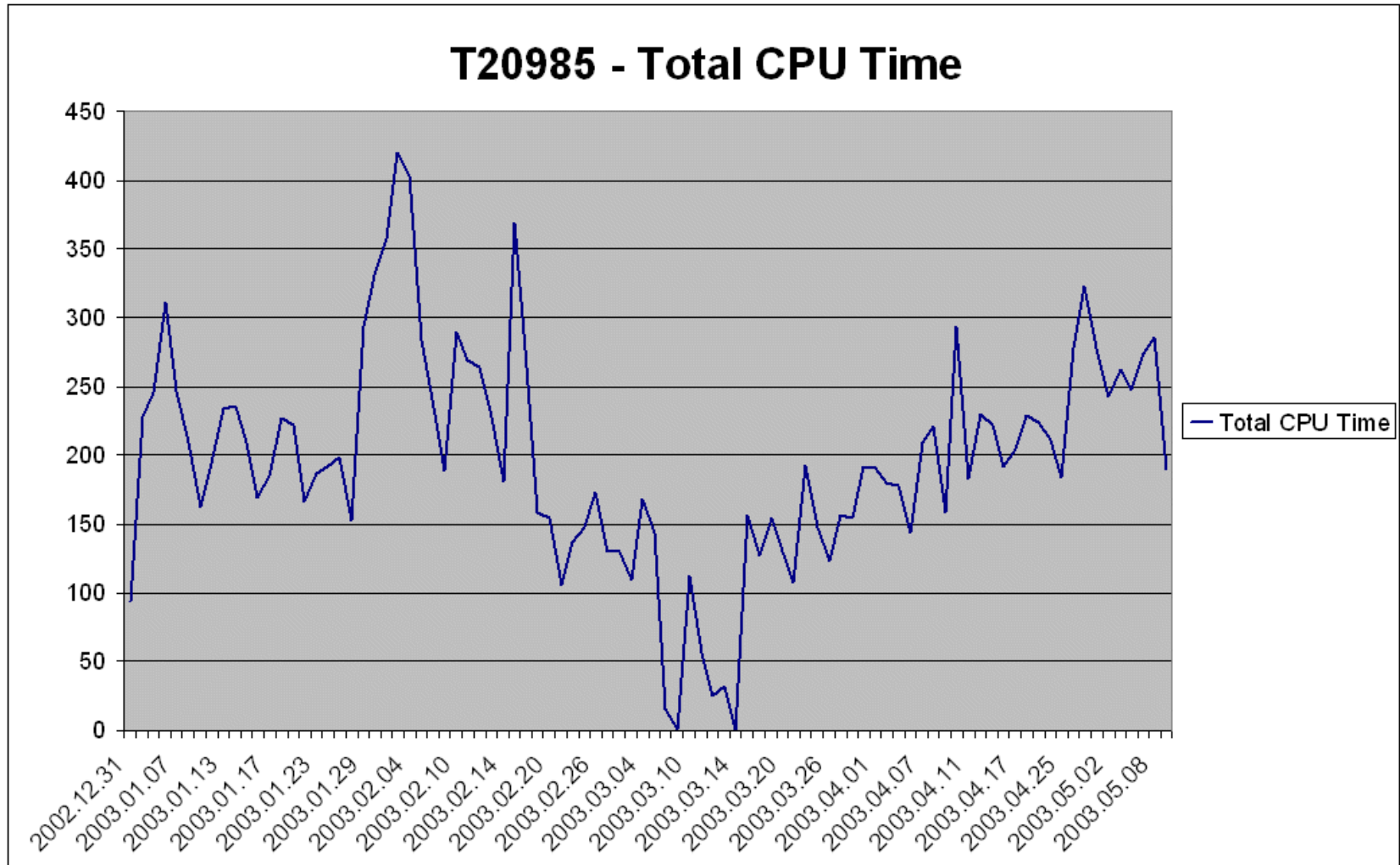


## Beispiel 4 – Datumsroutine unter C / LE

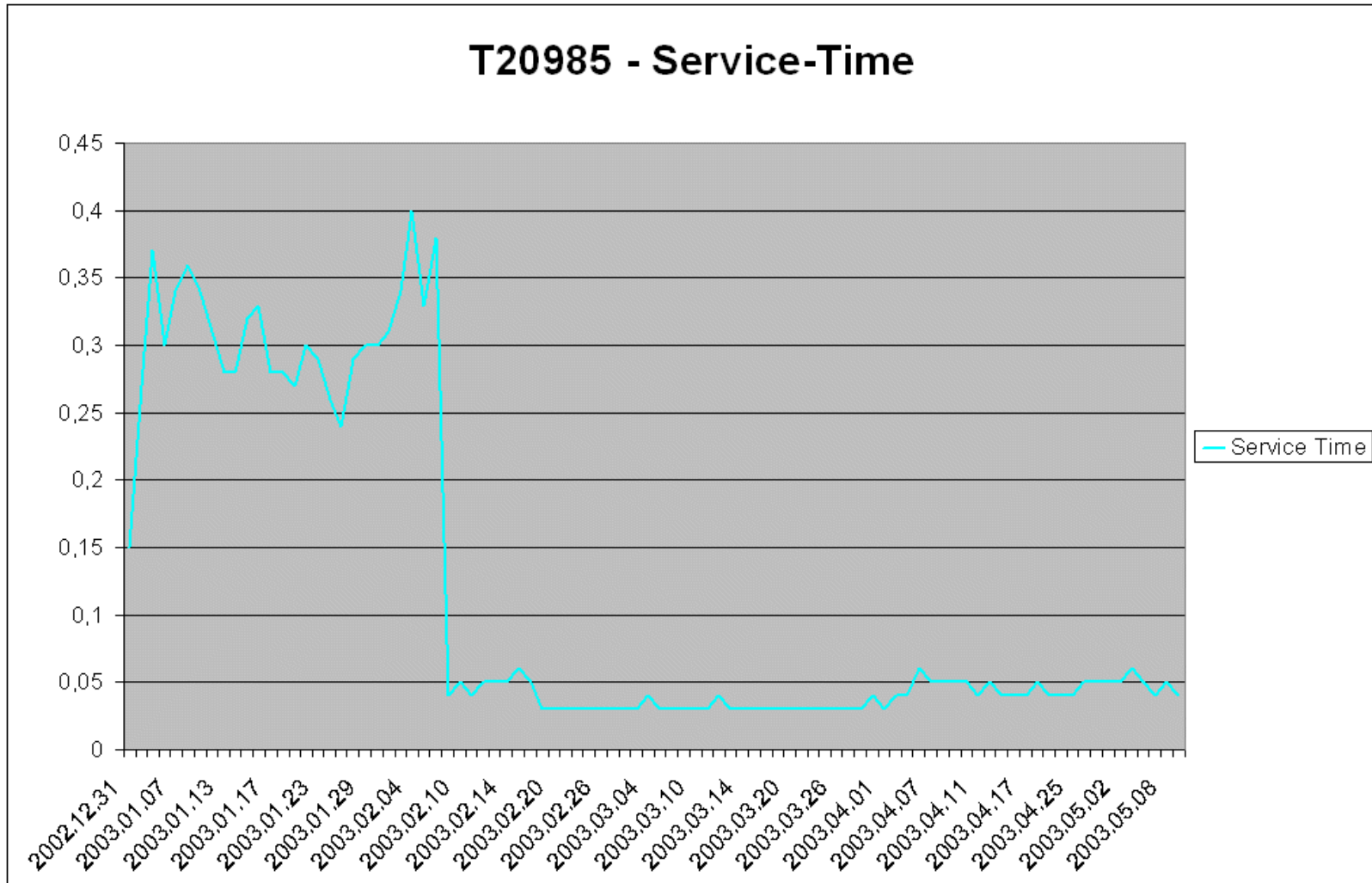
---

- „zufällig“ Anfrage wegen Routine
- mehr als 5 Mio. Aufrufe pro Tag
  - Auswertung(en) für Vorstand ☹
- GETMAIN / FREEMAIN
- LE-Enclave für C aufgebaut / abgebaut
- Optimierungsversuche (ca. ½ Jahr)
- Umschreiben auf COBOL brachte Erfolg
- „Einsparung“ ca. 3.500 CPU-Stunden p.a.
  - ca. 1,5 Prozessoren

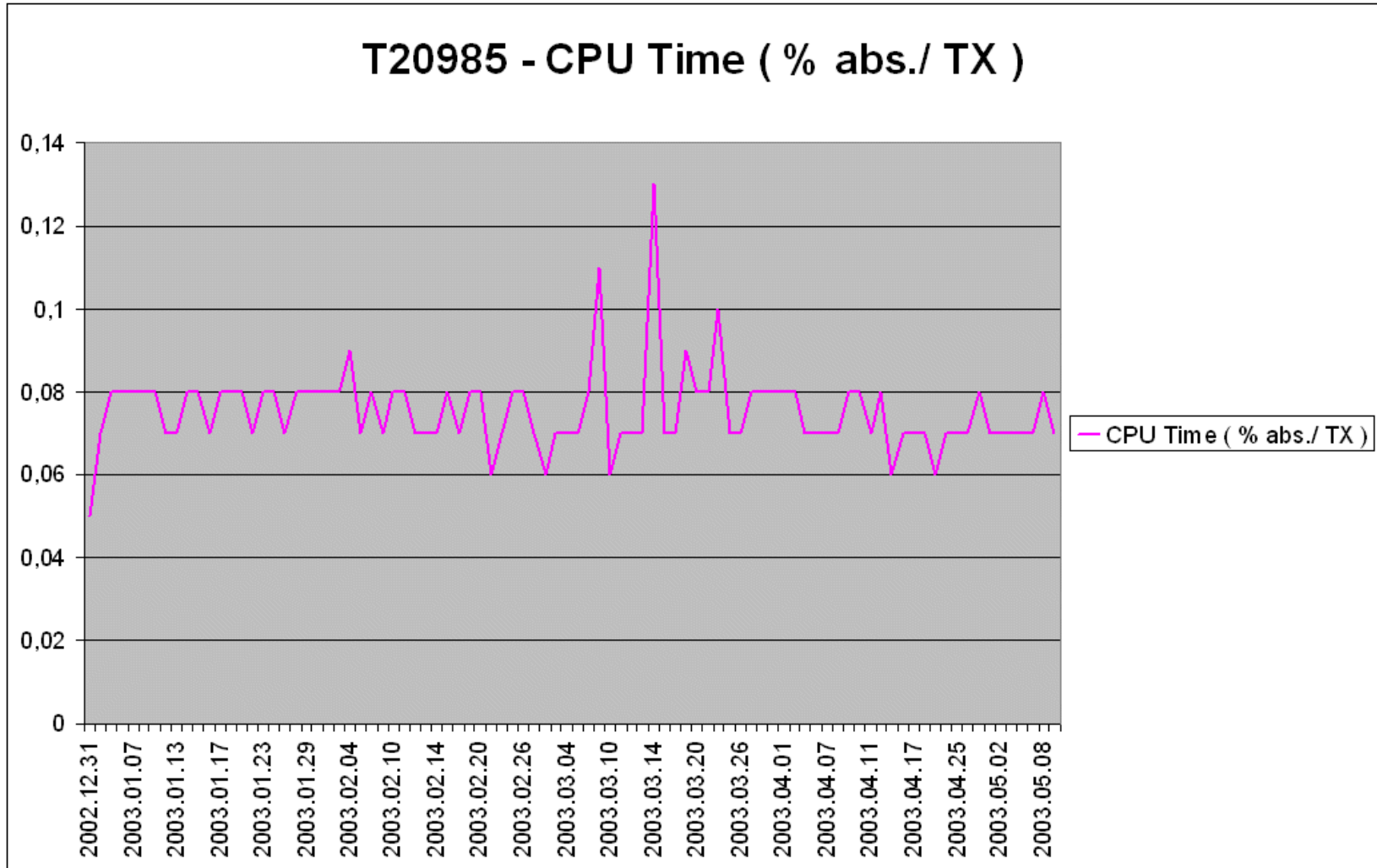
## Beispiel 5 – kein Handlungsbedarf – 1



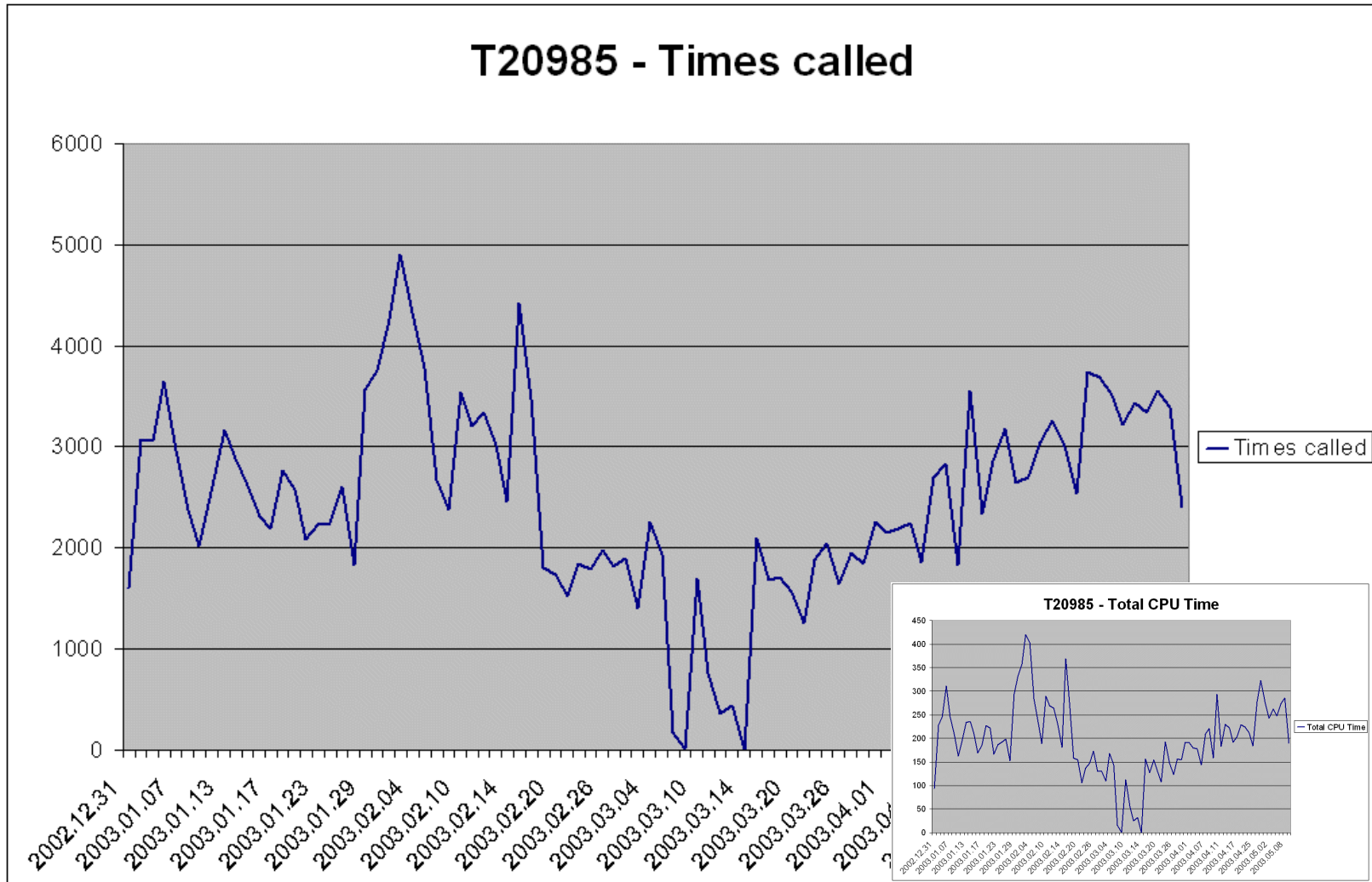
## Beispiel 5 – kein Handlungsbedarf – 2



## Beispiel 5 – kein Handlungsbedarf – 3



## Beispiel 5 – kein Handlungsbedarf – 4



# Optimierungen – Beispiele und Potential

## Beispiele – mögliche Erfolge – 1

Datum	Programm / Job	CPU Ersparnis (hochgerechnet auf 1 Jahr)	Laufzeit Ersparnis (hochgerechnet auf 1 Jahr)	Art der Optimierung
19.12.2002	I9L55	1000 Stunden	(Online)	versch.
03.12.2002	GTE71	300 Stunden	(Online)	Index eingeführt
05.11.2002	GTF83 / TGTF83*	1000 Stunden	3000 Stunden	DB2-Zugriff umkodiert
05.11.2002	ZFU23 / *	350 Stunden <sup>(4)</sup>	nicht berechnet	COBOL-Layer
05.11.2002	ZZH75 / TZZH75*	250 Stunden <sup>(5)</sup>	14 Stunden	Index eingeführt
30.10.2002	DHI70/TDHI70*	270 Stunden	380 Stunden	Aufrufhäufigkeit ZFU23
23.10.2002	DW215/TDWDW215	250 Stunden	300 Stunden	SQL
30.10.2002	KK-Abschluss <a href="#">Detailinformation</a>	nicht messbar <sup>(2) (3)</sup>	nicht messbar	versch.
12.09.2002	DHJ20/TDHIJ20*	170 Stunden	400 Stunden	fachliche Änderungen
10.09.2002	DHN91/TDHN91*	100 Stunden	100 Stunden	fachliche Änderungen
02.09.2002	DO692/PDODO692	700 Stunden	850 Stunden	Reorganisation DB2-Table
22.08.2002	I9S61 / IMS	240 Stunden	(Online)	Loop (Code).
15.08.2002	N1451 / TN1451*	270 Stunden	336 Stunden	Aufrufhäufigkeit N2U73
15.08.2002	ZZH75 / TZZH75*	800 Stunden <sup>(5)</sup>	0 Stunden	Parallelisierung / Fachlichkeit
14.08.2002	DM238 / TDM238*	400 Stunden	400 Stunden	Einsatz internes Array
08.08.2002	U6W89 / TU6W89*	900 Stunden	900 Stunden	Index eingeführt
07.08.2002	PK110 / PPKPK110	300 Stunden <sup>(2)</sup>	400 Stunden	Aufruf ZFU23
11.07.2002	GTF83 / TGTF83*	- - - (3)	- - - (3)	Runstats/Rebind
11.07.2002	KIA99 / IMS	55 Stunden	(Online)	Code-Optimierung
20.06.2002	HAA45 / THAHA23*	270 Stunden	450 Stunden	Index eingeführt
28.05.2002	DHV08 / TDHDHV08	90 Stunden	800 Stunden	SQL-Zugriff
29.04.2002	GTK00 / versch.	750 Stunden	1875 Stunden	SQL-Zugriff, RUNST. etc.
29.04.2002	ZFU88 / versch.	140 Stunden	200 Stunden	Felddefinitionen
23.04.2002	DHI25 / TDHDHI25	120 Stunden	200 Stunden	Initialize, DBRM etc.
27.03.2002	GTL01 / TGTL0151	420 Stunden	nicht berechnet	Inspect / Initialize
11.03.2002	HH720 / xHH72001	38 Stunden <sup>(5)</sup>	30 Stunden	COBOL-Felder (Stufe 1)
05.03.2002	HAA24 / THAHA24W	400 Stunden	400 Stunden	SQL-Zugriff
17.02.2002	TD1D1B30	- - -	40 Stunden	BUFNO=16
05.02.2002	GT500 / MGT500%1	- - - (3)	- - - (3)	RUNSTATS, REBIND
05.02.2002	BF001 / MBFBF001	240 Stunden	310 Stunden	DB2-Optimierung
28.01.2002	N2735 / TN2735*	625 Stunden	667 Stunden	Aufrufhäufigkeit I9U73
25.01.2002	Posy / TPOPO001	- - -	30 Stunden	BUFNO=16
22.01.2002	IK019 / TIKIK019	40 Stunden	120 Stunden	Aufrufhäufigkeit I9U73
		<b>3.317.559,12 € <sup>(1)</sup></b>		

## Beispiele – mögliche Erfolge – 2

Datum	Programm / Job	CPU Ersparnis (hochgerechnet auf 1 Jahr)	Laufzeit Ersparnis (hochgerechnet auf 1 Jahr)	Art der Optimierung
14.05.2003	ZFF01 / *	315 Stunden <sup>(4)</sup>	Online	COBOL-Tuning / Pgmlogik
21.03.2003	I9S76 / *	290 Stunden <sup>(4)</sup>	Online	DB2-Optimierung
05.03.2003	ZFU23 / *	400 Stunden <sup>(3)</sup>	nicht berechnet	COBOL
17.02.2003	<a href="#">IK008 / TIKIK008</a>	50 Stunden	60 Stunden	Sort, Programmlogik
11.02.2003	<a href="#">PAX05 / T33*, T08*</a>	940 Stunden	Online	DB2-Optimierung
11.02.2003	<a href="#">I9L55 / T21574</a>	1000 Stunden	Online	DB2-Optimierung
11.02.2003	N3A02 / TN2A02*	295 Stunden <sup>(2)</sup>	1040 Stunden	Ausbau überflüssiger Code
11.02.2003	<a href="#">DDS09 / TDDDDS09</a>	80 Stunden	400 Stunden	SQL optimiert
14.01.2003	DH6*	70 Stunden	100 Stunden	Aufrufhäufigkeit I9U73
		<b>930.038,40 € <sup>(1)</sup></b>		





## Potential allgemein - Compile Options (\*)

---

• AWO NOAWO	0%	bis	10%	/ -
• NUMPROC(PFD NOPFD)	1%	bis	20%	/ 3%
• NOOPT OPT(STD)	1%	bis	12%	/ 3%
• OPT(STD FULL)	0%	bis	80%	/ 1%
• NOSSRANGE SSRANGE	1%	bis	27%	/ -
• TEST NOTEST	20%	bis	200%	/ -
• TRUNC(BIN STD) (**)	15%	bis	78%	/ 40%
• TRUNC(OPT STD)	6%	bis	65%	/ -

\* IBM-Zahlen aus IBM Enterprise COBOL Version 3 Release 1 Performance Tuning, January 16, 2002 u.a.

\*\* IBM will TRUNC(BIN) optimieren ab/seit COBOL for OS/390 & VM V2R2: bisher wenig überzeugend -> **V5.1!!**





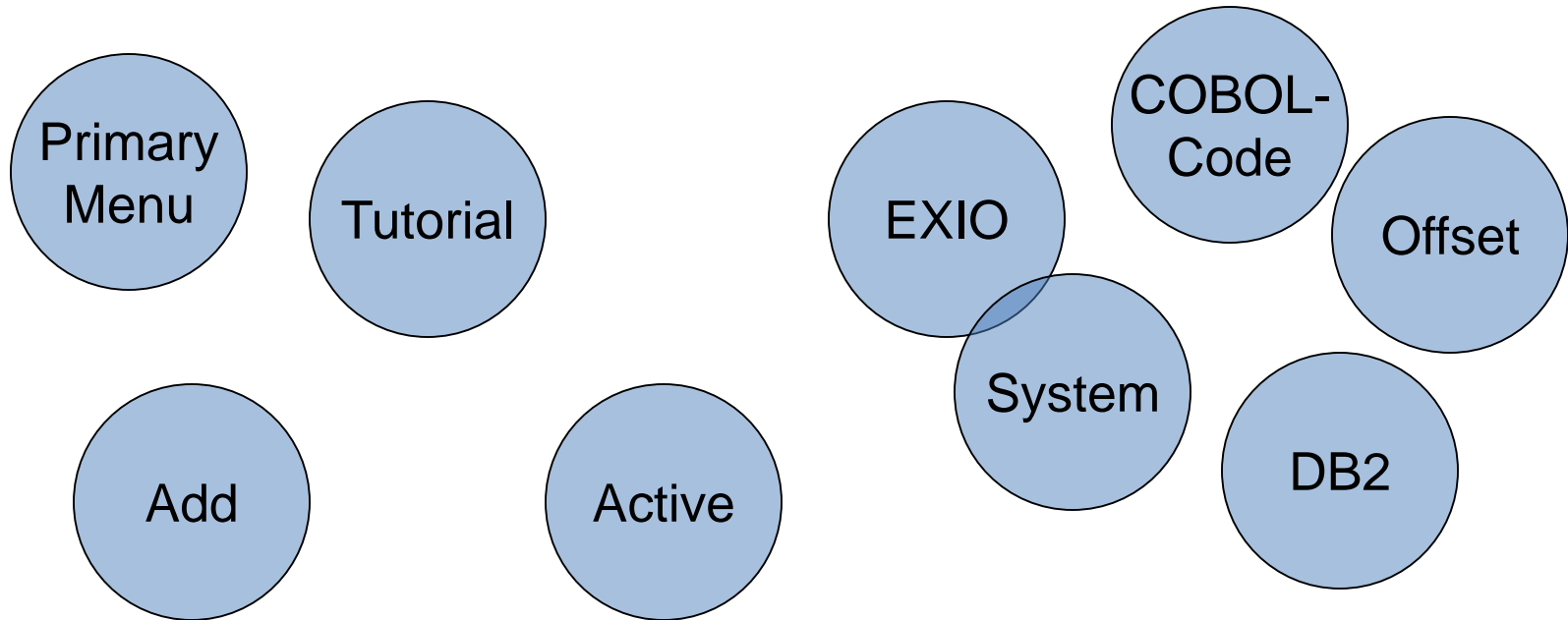
## Basis bei RRR

---

- Informationen zu DBRMs / Packages
- Top-100-Liste CICS (07:00 bis 17:00 Uhr)
- Performancebilder von Transaktionen
  
- laufende Beobachtung
- kein echtes Reporting über Erfolge



- 
- Vorstellung und Einführung
  - Optimierungen – Beispiele und Potential
  - ➔ • Strobe – Handling
  - Vorgehensweise für die Analyse (Strobe)
  - Vorgehensweise für die Analyse (iStrobe)
  - Modellierung und DB2-Zugriffe
  - COBOL–Felder – COBOL-Befehle
  - Auswirkungen von Optionen – COBOL – LE
  - Praxisbeispiele – Diskussion – Austausch

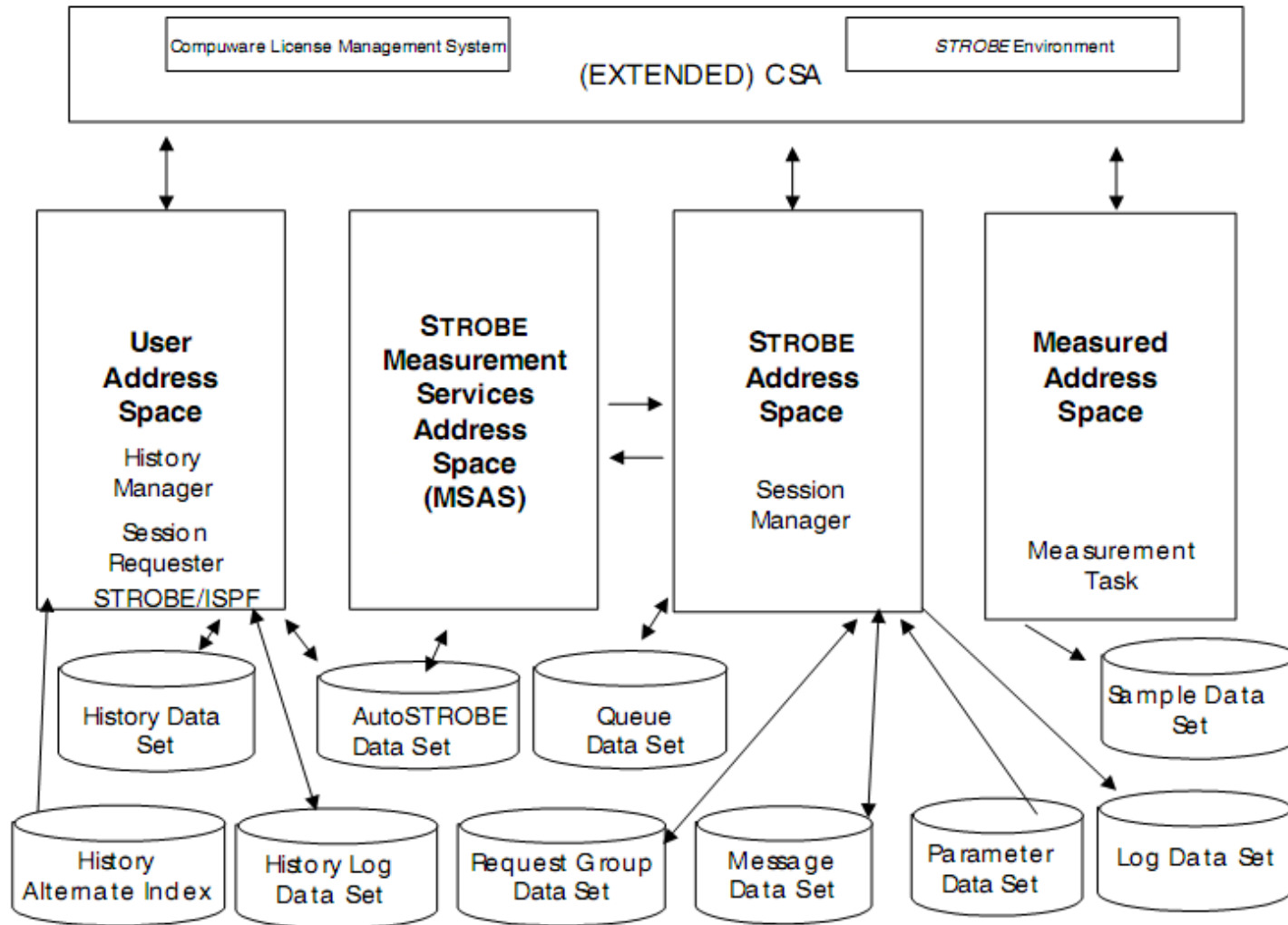


- Strobe zieht Bilder aus dem Adressraum
  - Samples
  - „Stroboskop“
- Informationen werden in Datei gesammelt
  - Sample Dataset
- Sample Dataset wird zu einem Profil aufbereitet
  - Profile Dataset
  - xml für iStrobe
- Source kann hinterlegt werden
  - Mapping / Indexing

- Strobe hat Erweiterungen für z/OS Subsysteme
  - Capture Options
  - DB2 (jetzt auch DDF) / IMS / MQ / Websphere
- Strobe unterstützt Programmiersprachen
  - COBOL, PL1, ASM, C/C++
- eigene Interfaces sind möglich
  - Data Collectors

# Strobe – Handling

## allgemeines / Übersicht



- TSO STROBE
- TSO %STROBE
- R.STR
- via Eclipse
  
- <http://istrobe.rrr.de>

# Strobe – Handling

## Hauptmenü

```
----- Strobe Options ----- Release 04.03.00
OPTION  ==>                                PTF LEVEL 043007

0  USER DEFAULTS      - Strobe/ISPF user default options
1  ADD ACTIVE          - Add a measurement request for an executing job
2  ADD QUEUED          - Add a measurement request for a job not yet executing
3  STATUS              - Monitor/change measurement requests and create profiles
4  PROFILE             - Create a Profile of a Strobe measurement session
5  INDEX               - Create a map data set
6  BATCH-SUBMISSION   - Batch submission of Strobe commands
M  MESSAGES            - Display information about a Strobe message
L  LOG UTILITY         - Perform Strobe log utility function
P  PACKAGING UTILITY  - Perform Strobe packaging utility function
T  TUTORIAL            - Display information about Strobe
C  CHANGES            - Display summary of changes in this release

AutoStrobe
G  REQUEST GROUPS     - Create, edit and submit groups of measurement requests

X  EXIT                - Terminate Strobe/ISPF
```



## 0 – User Defaults

----- Strobe - User Default Options -----

OPTION ==>

- 1 JOB STATEMENT - Specify template job statement
- 2 SYSOUT PARAMETERS - Specify SYSOUT parameters
- 3 TEMP DSN PARAMETERS - Specify TEMPORARY dataset parameters

## 0.1 – Job Statement

```
----- Strobe - Job Statement -----  
COMMAND ==>  
  
JOB STATEMENT INFORMATION:  
(Used to initialize job statements on all other panels)  
  
==> //XV8822DA JOB '1N289003010200100000', 'RALF.SEIDLER',  
==> //          MSGCLASS=Q, CLASS=S,  
==> //          NOTIFY=XV8822D      , REGION=0M  
==> // *MAIN ROOM=AL21-L2X
```

# Strobe – Handling

## 0.2 – Sysout Parameter

```
----- STROBE - SYSOUT PARAMETERS -----  
COMMAND ==>  
  
BACKGROUND INDEX PRINT OUTPUT  
  
    SYSPRINT ==> *  
    SYSUDUMP ==> *  
  
BACKGROUND PROFILE PRINT OUTPUT  
  
    PROFILE   ==> *  
    SORT      ==> *  
    SYSUDUMP  ==> *  
    SYSTEM    ==> *  
  
BATCH-SUBMISSION OUTPUT  
  
    CLASS     ==> *
```

## 0.3 – Temporary dataset parameters

```
----- Strobe - Temporary Dataset Parameter -----  
COMMAND ===>  
  
TEMPORARY DATASET PARAMETER:  
(Used to override the default high level qualifier (HLQ) for temporary  
dataset names created by STROBE/ISPF. Defaults to TSO userid or  
TSO profile prefix.)  
  
===> XV8822D
```

## T – Tutorial

TUTORIAL ----- TABLE OF CONTENTS -----

OPTION ==>

```
-----  
| STROBE/ISPF TUTORIAL |  
| TABLE OF CONTENTS |  
-----
```

0	User Defaults	-	Strobe/ISPF user default options
1	Add Active	-	Add a measurement request for an executing job
2	Add Queued	-	Add a measurement request for a job not yet executing
3	Status	-	Monitor measurement requests and create Profiles
4	Profile	-	Create a Strobe Performance Profile
5	Index	-	Create a map dataset
6	Batch-Submission	-	Submit a measurement request as a batch job
A	AutoStrobe	-	Add, delete and display AutoStrobe requests
G	Request Groups	-	Create, edit, or submit a group of requests
H	History	-	Access historical data
M	Messages	-	Display information about a Strobe message
L	Log Utility	-	Perform Strobe log utility functions
P	Packaging Utility	-	Perform Strobe packaging utility functions
C	Commands	-	Additional Strobe/ISPF commands
I	Introduction	-	Getting started with Strobe/ISPF
R	Reports	-	The Strobe Performance Profile

# Strobe – Handling

## M – Messages – Eingabe

```
----- STROBE - DISPLAY MESSAGE TEXT -----
```

```
COMMAND ==>
```

```
MESSAGE NUMBER--STR ==> 6801      (Numeric portion)
```

```
Enter the message number (the numeric portion of the message identifier)  
to display the text associated with the message.
```



## 6 – Batch-Submission

```
----- Strobe - Batch Submission Options -----  
OPTION ==>  
  
1  ADD ACTIVE      - Add a measurement request for an executing job  
2  ADD QUEUED     - Add a measurement request for a job not yet executing  
3  CHANGE ACTIVE  - Send a sampling control command to an active request  
4  CHANGE QUEUED  - Change a queued measurement request  
5  CHANGE COMPLETED - Change a completed measurement request  
6  LIST/DELETE    - Display/delete measurement requests  
7  JOB STREAM MENU - Process the generated job stream  
8  CHANGE DEF/ACTIVE - Change a deferred active measurement request  
9  CHANGE DEF/QUEUED - Change a deferred queued measurement request  
G  SUBMIT A GROUP - Submit a request group  
  
END  CANCEL      - Exit without submitting the generated job stream
```



# Strobe – Handling

## 5 – Indexing / Mapping (nicht bei RRR)

```
----- STROBE - INDEX TO CREATE A MAP DATA SET -----  
OPTION ===>  
  
      B - Background processing    F - Foreground processing  
  
PROGRAM LANGUAGE: (Specify Y -- select one language)  
ASSEMBLER      ===>      C (IBM OR SAS)  ===>      COBOL          ===>  
PL/I           ===>  
  
OUTPUT: MAP DATA SET  
      ===>  
UNIT ===> SYSDA      VOLUME ===>  
  
INPUT: COMPILER SYSPRINT DATA SETS  
      ===>  
      ===>  
      ===>  
      ===>  
      ===>
```

## 4 – Profile – Performance Profil erzeugen

```
----- Strobe - Produce a Performance Profile -----
OPTION ==>

      B - Background processing      F - Foreground processing

ENTER BLANKS TO VIEW A DATASET SELECTION LIST
SAMPLE DATASET NAME  ==> 'USER.STROBE.T03N661.S001D001.STROBE'
      UNIT ==>                VOLUME ==>
SPECIFY PROFILE REPORT PARAMETERS: (Y or N)
  Detail Reports      ==> N    Tailor Reports      ==> N    Indexing      ==> N

OUTPUT FORMAT        ==> N    W -Wide Report      I -iStrobe File
                        N -Narrow Report

NUMBER OF COPIES FOR BACKGROUND REPORTS ==>

Specify a dataset name to save a copy of the Strobe Profile Report:
DATASET  NAME ==>
      UNIT ==> SYSDA          VOLUME ==>

Specify a SYSIN dataset containing parameters for the Reporter:
DATASET  NAME ==>
```

# Strobe – Handling

## 3 – Status – Überblick über die Messungen

```
----- STROBE - STATUS -----
COMMAND ==>                                SCROLL ==> PAGE

SYSTEM      ==> S51      (System or clear to select or *ALL for all systems)
OWNERID MASK ==> *      (Enter * for a pattern match)

  V =View    C =Change  P =Profile Report      G =default Profile
  Q =Quit    D =Delete  X =Delete with data sets  B =Browse Auto Profile
                                           MORE>>

----- ACTIVE MEASUREMENT REQUESTS -----
---REQ-OWNERID--SYSTEM---JOBNAME--PROGRAM-STEPNAME/STEPNUM-TARGET--TOTAL---EXEC
1291 X016291  S51      X016291          SPFCUA.SPFCUA      2000    292    0

----- QUEUED MEASUREMENT REQUESTS -----
---REQ-OWNERID--SYSTEM---JOBNAME--PROGRAM-STEPNAME/STEPNUM-TARGET-DURATION-RESP
1292 X016291  S51      X016291          JU44760.IMS      2000          2 CONT

----- COMPLETED MEASUREMENT REQUESTS -----
---REQ-OWNERID--SYSTEM---JOBNAME--PROGRAM-STEPNAME/STEPNUM-TARGET--TOTAL---EXEC
1276 C147060  S51      C7061672          *OMVSEX          10000  10000  10000
1277 C147060  S51      X011516B          UACC02           10000  10000  10000
1278 C147060  S51      ICMRR51           LOAD            10000   3411   3411
1279 C479020  S51      ICMRR51           LOAD            10000   9299   9299
1280 ICSTRONL S51      CICSTK            CICSTK          30000  30000   809
```

# Strobe – Handling

## 1 – Add Active

```
----- Strobe - Add Active Request -----  
COMMAND ==>  
  
JOBNAME ==> cicspa0j (Jobname or clear to list active Jobs)  
  
SYSTEM ==> mvsc (System or clear to list available systems)  
SCHEDULE REQUEST ==> N (Y or N; Use Y only when overriding the default)  
AUTO PROFILE CREATION ==> N (Y or N; Use Y only when overriding defaults)  
  
MEASUREMENT SESSION INFORMATION:  
SESSION DURATION ==> 1 (Estimated time in minutes)  
TARGET SAMPLE SIZE ==> 10000 (Target number of samples)  
  
TSO USERID TO NOTIFY ==> XV8822D (Notify when session completes)  
  
SAMPLE DATASET INFORMATION:  
DATASET NAME PREFIX ==> USER.STROBE  
UNIT NAME ==> SYSDA VOLUME ==> DISP ==> CATLG (CATLG OR KEEP)  
  
SELECT ADDITIONAL PARAMETERS: (Y or N; Use Y only when overriding defaults)  
DATA COLLECTORS ==> N MODULE MAPPING DATA ==> N  
SESSION MANAGEMENT ==> N REQUEST RETENTION ==> N  
OTHER PARAMETERS ==> N SAVE REQUEST IN GROUP ==> N
```

# Strobe – Handling

## 2 – Add Queued

```
----- Strobe - Add Queued Request -----  
COMMAND ==>  
JOBNAME   ==> xv8822d1 (Jobname or jobname* for pattern match)  
PROGRAM   ==>          (Program or clear both PROGRAM & STEP to input a list)  
STEP      ==> *ALL     (Name,num,step.procstep or *ALL for all steps)  
SYSTEM    ==> *ALL     (System or clear for a list or *ALL for all systems)  
SCHEDULE REQUEST ==> N (Y or N; Use Y only when overriding the default)  
AUTO PROFILE CREATION ==> N (Y or N; Use Y only when overriding defaults)  
  
MEASUREMENT SESSION INFORMATION:  
SESSION DURATION   ==> 10      (Estimated time in minutes)  
TARGET SAMPLE SIZE ==> 20000   (Target number of samples)  
  
TSO USERID TO NOTIFY ==> XV8822D (Notify when session completes)  
  
SAMPLE DATASET INFORMATION:  
DATASET NAME PREFIX ==> USER.STROBE  
UNIT NAME   ==> SYSDA   VOLUME ==>          DISP ==> CATLG (CATLG OR KEEP)  
  
SELECT ADDITIONAL PARAMETERS: (Y or N; Use Y only when overriding defaults)  
DATA COLLECTORS           ==> N      MODULE MAPPING DATA           ==> N  
SESSION MANAGEMENT       ==> N      REQUEST RETENTION              ==> N  
OTHER PARAMETERS         ==> N      SAVE REQUEST IN GROUP          ==> N
```

# Strobe – Handling

## 1/2 – zusätzliche Angaben – Schedule Request

```
----- Strobe - Request Scheduler -----  
COMMAND ==>  
  
JOBNAME:      XV8822D1                      Specify dates in DD/MM/YYYY format  
                                           Current Date: 09/02/2014  
Select day of week  MON  TUE  WED  THU  FRI  SAT  SUN  Time: 11:54  
by entering a Y  ==>  _  _  _  _  _  _  _  
  
Starting Time(s) ==> 1) __ : __  24HR  2) __ : __  24HR  3) __ : __  24HR  
Starting Date    ==> __ / __ / ____  
Number of Weeks  ==> __ (1-52)  OR  Ending Date    ==> __ / __ / ____  
  
Add or exclude dates and times generated above or create a customized list  
(OPT: A - Add a date and time      E - Exclude a date and time)  
  
OPT      DATE          TIME  24HR      OPT      DATE          TIME  24HR  
_  _ / _ / ____  _ : _      _  _ / _ / ____  _ : _  
_  _ / _ / ____  _ : _      _  _ / _ / ____  _ : _  
_  _ / _ / ____  _ : _      _  _ / _ / ____  _ : _  
_  _ / _ / ____  _ : _      _  _ / _ / ____  _ : _  
_  _ / _ / ____  _ : _      _  _ / _ / ____  _ : _  
_  _ / _ / ____  _ : _      _  _ / _ / ____  _ : _
```

## 1/2 – zusätzliche Angaben – Data Collectors

```
----- Strobe - Data Collectors -----
```

```
COMMAND ==>
```

```
OVERRIDE DATA COLLECTOR DEFAULTS FOR JOBNAME:  XV8822D1
```

Strobe will attempt to collect data for all of the licensed Data Collectors.  
Tailor the collection of data by selecting specific Data Collectors.  
Collector name followed by a plus sign(+) indicates additional options.

```
DATA COLLECTORS: (Y to select or N to deselect specific collectors)
```

```
CICS +           ==> N   COBOL           ==> Y   DB2 +           ==> Y  
IMS +           ==> Y   MQSERIES +      ==> Y   SVC             ==> Y  
VSAM            ==>
```

```
OTHER DATA COLLECTORS:
```

```
PROGRAM NAME    ==>           ==>           ==>           ==>
```

## 1/2 – zusätzliche Angaben – Data Collectors – Pop Ups

```
----- Strobe - DB2/IMS/DDF Data Collector Capture Option -----
```

```
COMMAND ==>
```

```
Based on Data Collectors:   DB2 = Y   IMS = Y
```

```
Please specify CAPTURE Options:
```

```
DB2 ==>   (Y/F/N)   IMS ==>   (Y/N)
```

```
DB2 Report Type ==> B (P/A/B) Performance/Accounting/Both
```

```
-----
```

```
----- Strobe - MQ Series Data Collector Options -----
```

```
COMMAND ==>
```

```
MQ Common User Module ==>
```

```
Always use as default ==>   (Y/N)
```



## 1/2 – zusätzliche Angaben – Module Mapping Data

```
----- STROBE - MODULE MAPPING -----  
COMMAND ==>  
  
ADDITIONAL MEASUREMENT INFORMATION FOR JOBNAME:  X016291  
  
LIBRARIES TO SEARCH to get module mapping data:  
    ==>  
    ==>  
    ==>  
  
DETAIL BASELINE    ==> 2                (Minimum percent of time spent executing  
                                         a module for it to be mapped)  
  
BASELINE OVERRIDE ==>                ==> (Names of load modules to be  
                                         mapped without regard to the  
                                         BASELINE restriction)  
    ==>                ==>  
    ==>                ==>  
  
SVC NUMBERS    ==>                ==> (Specify SVCs--a number or range  
                                         of numbers (e.g., 1-5)--whose  
                                         modules are to be identified)  
    ==>                ==>
```

## 1/2 – zusätzliche Angaben – Session Management Parameters

```
----- Strobe - Session Management Parameters -----
```

```
COMMAND ==>
```

```
ADDITIONAL MEASUREMENT INFORMATION FOR JOBNAME:  XV8822D1
```

```
NUMBER OF SESSIONS  ==> 1      (The number of consecutive times the target  
                                sample size is to be collected.  Each  
                                session creates a new sample data set.)
```

```
FINAL SESSION ACTION ==> Q      (Action occurs at end of the last session.  
                                (Q)uit--Stop sampling, close sample file.  
                                Measurement session is terminated.  
                                (S)top--Stop sampling, close sample file.  
                                Measurement session remains active.  
                                (C)ontinue sampling--Do not take any action  
                                at target sample size. Number of  
                                sessions must be 1.)
```

# Strobe – Handling

## 1/2 – zusätzliche Angaben – Request Retention

----- Strobe - Request Retention -----

COMMAND ==>

ADDITIONAL INFORMATION FOR JOBNAME: XV8822D1

Today's date is 09/02/2014 (DD/MM/YYYY)

QUEUED RETENTION ==> (Specify the number of days from today that  
the queued request should be deleted.)

COMPLETED RETENTION ==> (Specify the number of days to retain  
the completed request.)

# Strobe – Handling

## 3 – Status / 4 Profil erzeugen

```
----- Strobe - Status -----
COMMAND ==>                                SCROLL ==> CSR

SYSTEM      ==> *ALL      (System or clear to select or *ALL for all systems)
OWNERID MASK ==> XV8822D* (Enter * for a pattern match)
CHANGE THRESHOLDS ==> N   (Y or N; to classify completed requests important)
  V =View    C =Change    P =Profile Report      G =default Profile
  Q =Quit    D =Delete    X =Delete with data sets B =Browse Auto Profile
                                                    MORE>>

----- QUEUED MEASUREMENT REQUESTS -----
---REQ-OWNERID--SYSTEM---JOBNAME--PROGRAM-STEPNAME/STEPNUM-TARGET-DURATION-RESP
1318 XV8822D  *ALL      XV8822D1          *ALL              20000          10 CONT
----- COMPLETED MEASUREMENT REQUESTS -----
---REQ-OWNERID--SYSTEM---JOBNAME--PROGRAM-STEPNAME/STEPNUM-TARGET--TOTAL--EXEC
P 1317 XV8822D  MVSC      CICSPA0J          CICSPA0J          10000  10000      3
```

# Strobe – Handling

## Profil erzeugen – Host / Hostdatei

```
----- Strobe - Produce a Performance Profile -----
OPTION ==> b

      B - Background processing      F - Foreground processing

ENTER BLANKS TO BYPASS THIS DATASET
SAMPLE DATASET NAME  ==> 'USER.STROBE.CICSPA0J.S001D001.STROBE'
                UNIT  ==> SYSDA                VOLUME ==> PRI563

SPECIFY PROFILE REPORT PARAMETERS: (Y or N)
  Detail Reports      ==> Y   Tailor Reports      ==> N   Indexing      ==> N

OUTPUT FORMAT          ==> N   W -Wide Report      I -iStrobe File
                        N -Narrow Report

NUMBER OF COPIES FOR BACKGROUND REPORTS ==>

Specify a dataset name to save a copy of the Strobe Profile Report:
DATASET NAME ==> >> zusätzlich Dateiangabe möglich <<
                UNIT ==> SYSDA                VOLUME ==>

Specify a SYSIN dataset containing parameters for the Reporter:
DATASET NAME ==>
```

# Strobe – Handling

## Profil erzeugen – Detail Reports

```
----- Strobe - Detail for a Performance Profile -----  
COMMAND ==>  
  
SAMPLE DATASET: 'USER.STROBE.CICSPA0J.S001D001.STROBE'  
  
REPORT OPTIONS:  
  TITLE      ==>  
  
  COMPRESS   ==>      =.      ==>      =.  
             ==>      =.      ==>      =.  
             ==>      =.      ==>      =.  
             ==>      =.      ==>      =.  
             ==>      =.      ==>      =.  
             ==>      =.      ==>      =.  
  
  DETAIL     ==>      ==>      ==>      ==>  
             ==>      ==>      ==>      ==>  
             ==>      ==>      ==>      ==>  
  
  RESOLUTION ==> 0004      SORT SIZE ==>      LINES/PAGE ==> 60  
  
OTHER PARAMETERS ==>
```

# Strobe – Handling

## Profil erzeugen – Ausgabe in EJES / Beta 92

```
DSSTAT - XV8822DS JOB03278 - 871 Records ----- Row 1 of 7
Cmd DDName      StepName ProcName Que* C Pri Cpy Hld Destination      Records
ccc ccccccccc/cccccccc ccccccccc cccc c ccc ccc ccc cccccccccccccccccccc cccccccccccc>
  JESJCLIN              TXT                      0
  JESMSG LG             HLD  Q   4   1     LOCAL          17
  JESJCL                HLD  Q   4   1     LOCAL          89
  JESYSMSG              HLD  Q   4   1     LOCAL         141
  SYSIN      PROFILE    TXT                      1
  SYSOUT     PROFILE    HLD  Q   4   1     LOCAL          50
s  PROFILE  PROFILE    HLD  Q   4   1     LOCAL         573
***** Bottom of Data *****
```

# Strobe – Handling

## Profil erzeugen – iStrobe

```
----- Strobe - Produce a Performance Profile -----  
OPTION ==> b  
  
    B - Background processing    F - Foreground processing  
  
ENTER BLANKS TO BYPASS THIS DATASET  
SAMPLE DATASET NAME  ==> 'USER.STROBE.CICSPA0J.S001D001.STROBE'  
        UNIT  ==> SYSDA          VOLUME ==> PRI563  
  
SPECIFY PROFILE REPORT PARAMETERS: (Y or N)  
Detail Reports      ==> Y    Tailor Reports      ==> N    Indexing      ==> N  
  
OUTPUT FORMAT          ==> I    W -Wide Report      I -iStrobe File  
                        N -Narrow Report  
NUMBER OF COPIES FOR BACKGROUND REPORTS ==>  
  
Specify a dataset name to save a copy of the Strobe Profile Report:  
DATASET NAME ==> >> zusätzlich Dateiangabe möglich <<  
        UNIT ==> SYSDA          VOLUME ==>  
Specify a SYSIN dataset containing parameters for the Reporter:  
DATASET NAME ==>
```



# Strobe – Handling

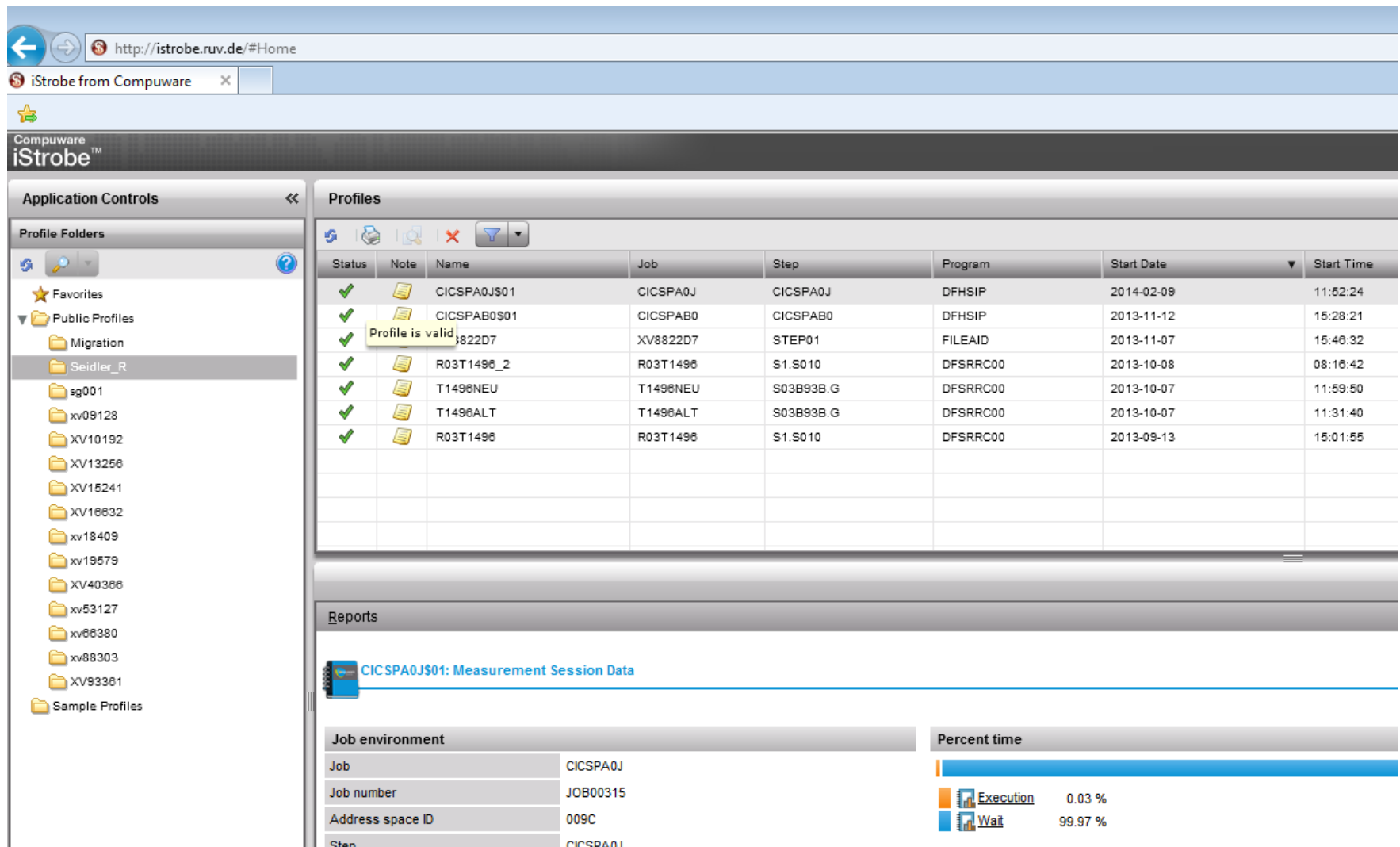
## Profil erzeugen – iStrobe – Definiton Zieldirectory

```
----- Strobe - iStrobe Performance Profile Options -----  
COMMAND ==>  
  
SAMPLE DATASET: 'USER.STROBE.CICSPA0J.S001D001.STROBE'  
  
iStrobe Data File ==> 'USER.STROBE.CICSPA0J.X001D001.STROBE'  
UNIT ==> VOLUME ==>  
  
SQL ANALYSIS FEATURE REPORTS ==> N (Y OR N)  
  
DOWNLOAD PARAMETERS:  
Name ==> CICSPAB0$01 Overwrite existing file ==> Y (Y or N)  
Description ==>  
Code Page ==> IBM-STD  
  
Perform Profile Transfer ==> Y (Y or N)  
iStrobe Manager:  
Server ==> istrobe  
Port number ==> 24354  
Folder ==> Public Profiles/Seidler_R  
  
Email Notification Address (optional):  
==>
```



# Strobe – Handling

## Profil erzeugen – iStrobe – Profil in iStrobe



The screenshot shows the iStrobe web application interface. The browser address bar displays <http://istrobe.ruv.de/#Home>. The page title is "iStrobe from Compuware". The main content area is divided into two sections: "Application Controls" and "Profiles".

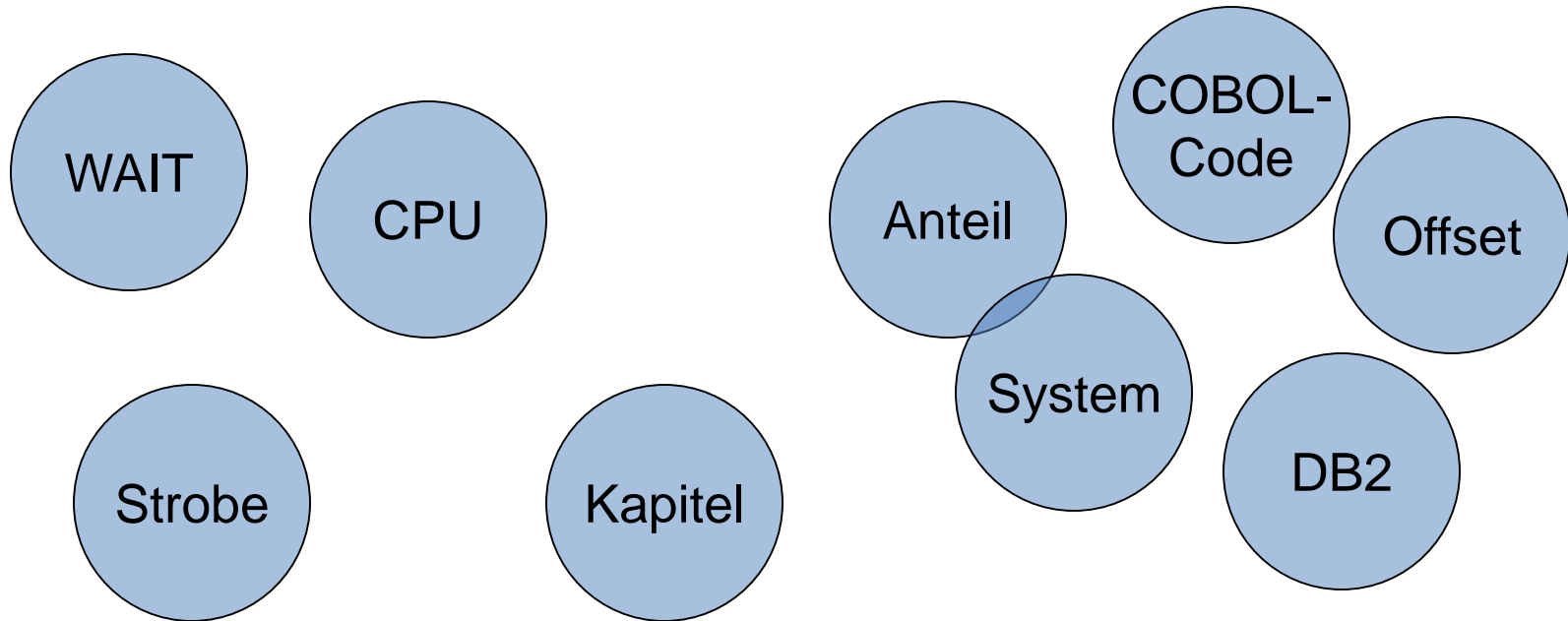
**Application Controls**

- Profile Folders
  - Favorites
  - Public Profiles
    - Migration
    - Seidler\_R
    - sg001
    - xv09128
    - XV10192
    - XV13258
    - XV15241
    - XV18632
    - xv18409
    - xv19579
    - XV40366
    - xv53127
    - xv66380
    - xv88303
    - XV93361
  - Sample Profiles

- 
- Vorstellung und Einführung
  - Optimierungen – Beispiele und Potential
  - Strobe – Handling
  - ➔ • Vorgehensweise für die Analyse (Strobe)
  - Vorgehensweise für die Analyse (iStrobe)
  - Modellierung und DB2-Zugriffe
  - COBOL–Felder – COBOL-Befehle
  - Auswirkungen von Optionen – COBOL – LE
  - Praxisbeispiele – Diskussion – Austausch

## Begriffe

---



## Ziel des Kapitels

---

- Das vorliegende Kapitel will versuchen, an Hand von konkreten Beispielen den Weg der Analyse zu beschreiben. Ziel ist es, so genannte „Eye Catcher“, d.h. offensichtliche Fehler, zu beleuchten. Diese treten in gut 90% aller Fälle auf. Für spezielle Analysen sollten stets Spezialisten hinzu gezogen werden.
- Das Kapitel beinhaltet Auszüge aus Messungen. Es wurden nur die relevanten Kapitel bzw. Kapitelteile aus den Messungen aufgenommen. Hinweise sind mit einem → gekennzeichnet und umrahmt.

## Datei – 1

Date: 2003.07.06 Job: WN281409 N2814 IKJEFT01

Chapter : #MSD

```
----- JOB ENVIRONMENT -----          ----- MEASUREMENT STATISTICS -----
PROGRAM MEASURED   -   IKJEFT1B          CPS TIME PERCENT   -   11.18
JOB NAME           -   WN281409          WAIT TIME PERCENT  -   88.82
JOB NUMBER        -   JOB18936          RUN MARGIN OF ERROR PCT -   .94
STEP NAME         -   N2814.IKJEFT01     CPU MARGIN OF ERROR PCT -   2.81
DATE OF SESSION   -   07/06/2003        TOTAL SAMPLES TAKEN -   20,877
TIME OF SESSION   -   17:52:06          TOTAL SAMPLES PROCESSED - 10,876
CONDITION CODE    -   C-0000            INITIAL SAMPLING RATE- 16.67/SEC
                                                           FINAL SAMPLING RATE  -   8.33/SEC

SYSTEM -          z/OS   01.03.00
DFSMS           -          1.3.0        SESSION TIME -    21 MIN 47.72 SEC
SUBSYSTEM       -          DB2 7.1.0    CPU TIME     -     2 MIN  9.89 SEC
DB2 SUBSYSTEM ID -          D205        WAIT TIME    -    17 MIN 11.85 SEC
DB2 APPLICATION -          N2814        STRETCH TIME -     2 MIN 25.97 SEC
CPU MODEL       -          2064-116
SYSTEM ID       -          P005        SRB TIME     -     0 MIN  4.54 SEC
LPAR            -          P005        SERVICE UNITS-    1,054,317
```

→ WAIT-Time ist überproportional hoch

→ WAIT-Analyse erforderlich

## Datei – 2

Chapter : #TDA

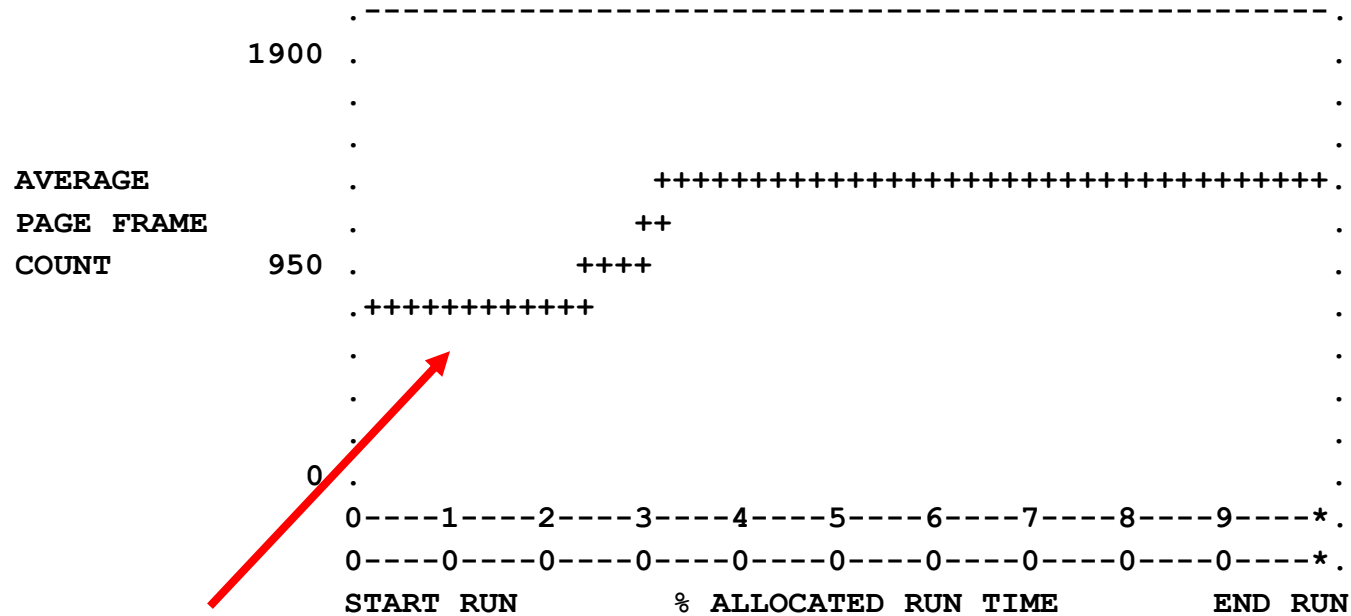
TASK OR DDNAME	RESOURCE	N*10 = % FULL UTILIZATION;	* IS > 95%;	- IS < 5%								
DSNECP10	CPU	.-	-132333333323221223-	-111112222211.								
DSN	CPU	.		.								
END	CPU	.		.								
N2UMSO2	3490	.	24666767898999989995	2889998899***.								
N2CONI4	3490	.	27553364333533264236	233222332111.								
N2UMSI2	3490	.	24566656455443423442	1334333343322.								
N2CONO5	3490	.	14232243122321132123	-422111221-1-.								
. FILEMGT		.	-	- - -.								
N2TXTO3	3490	.	1-21111-111111--112	-11112112----.								
N2TXTI3	3490	.	--11-11-1111--1---1	- -1-1-11- - .								
N2AUSO4	3490	.	--- - - - -	-- ---- .								
		0----	1----	2----	3----	4----	5----	6----	7----	8----	9----	*.
		0----	0----	0----	0----	0----	0----	0----	0----	0----	0----	*.
		START RUN	% ALLOCATED RUN TIME	END RUN								

- ➔ 30% der Laufzeit zu Beginn wird nichts getan
- ➔ 5% der Laufzeit ab 70% wird nichts getan
- ➔ Verursacher?



## Datei – 3

Chapter : #WSS



→ 30% der Laufzeit zu Beginn wird nichts getan  
→ Verursacher?

# WAIT-Analyse

## Datei – 4

Chapter : #WTM

MODULE	SECTION	COMPRESSED	FUNCTION	% RUN	TIME	MARGIN	OF ERROR	.94%
NAME	NAME			PAGE	TOTAL	00	19.00	38.00
.DB2	DSNVSR		SUSP/RES/CANCE	.00	2.15	++		
.IOCS	IGG019AQ		QSAM GET NEXT	.00	15.05	+++++++		
.IOCS	IGG019AR		QSAM PUT NEXT	.00	33.70	+++++++		
.IOCS	IGG019CW		SAM EOB CHAIN.	.00	.32			
				----	-----			
.IOCS	TOTALS		DATA MANAGEMENT	.00	49.07			
.NUCLEUS	IEAVESLL		SUSPEND LOCK S	.00	.07			
.SVC	SVC 001		WAIT	.00	37.50	+++++++		
.SVC	SVC 006		PROGRAM MANAGE	.00	.01			
.SVC	SVC 119		TESTAUTH	.00	.01			
				----	-----			
.SVC	TOTALS		SUPERVISOR CONTR	.00	37.52			

- ➔ Verursacher der WAITs:  
Datei lesen, Datei schreiben und expliziter WAIT
- ➔ expliziter WAIT passt zu 30% am Beginn des Jobs
- ➔ Joblog-Analyse ergibt in diesem Fall:  
das war ein WAIT auf Kasette

## Datei – 6

---

Chapter : #SWS

SQL	SQL	STMT-EXECUTION	TIME/CNT	% RUN	TIME	MARGIN	OF ERROR	.94%
TYPE	NAME	CNT	AVG-TIME	PAGE	TOTAL	00	4.00	8.00
DBRM	N2814	32,816	.0010	13:15:40	1.08	1.84	****+	
				-----	-----			
TOTAL WAIT ACTIVITY				1.31	2.13			

**→ Nichts Außergewöhnliches**

## Datei – 7

---

Chapter : #WBS

DBRM - N2J36

CREATED - 11/23/1999 11:06:38

STATIC, NON-CURSOR SQL

```
3138 SELECT MANDANTK, ZAUFRID, ZVUMLFNR, STUMLBEA, KTOINHNR, KTONR, ISOWHRCD, BUCHDA
      T, UABUKTOU, UMSUMART, UMSSPTS
      INTO :H, :H, :H, :H, :H, :H, :H, :H, :H, :H, :H FROM RWBUUIVI WHERE MANDANTK=:H AN
      D KTOINHNR=:H AND KTONR=:H AND ISOWHRCD=:H AND BUCHDAT=:H AND UMSSPTS=:H
```

STMT NUMBER	STATEMENT TEXT	STMT-EXECUTION CNT	AVG-TIME	% RUN TIME PAGE	TIME TOTAL	MARGIN OF ERROR 00	1.00	.94% 2.00
3138	SELECT			.09	.13	**		
		19,128	.0001					
		-----	-----	-----	-----			
	TOTALS	19,128	.0001	.09	.13			

**→ Nichts Außergewöhnliches**

# WAIT-Analyse

## Datei – 8

Chapter : #ACW

```
-----INVOKED BY-----          -----VIA-----          -WAIT TIME%-
XACTION  MODULE  SECTION  RETURN  LINE  MODULE  SECTION  PAGE  TOTAL
.DB2     DSNVSR                SUSP/RES/CANCEL SYNCHRO

XACTION  QUERY NAME          TIME          TEXT          STMT  TEXT          PAGE  TOTAL
N2814    N2814                13:15:40      5387 OPEN          .00  1.24
-----
                          .00  2.15

.SVC     SVC 001                WAIT

          N2814    N2814    007962          IGZEQOC          .00  22.85
          N2814    N2814    007B8E          IGZEQOC          .00  2.50
          N2814    N2814    007D9E          IGZEQOC          .00  3.74
N2814    .IOCS    IGG019AQ          QSAM G    SVC 055          .00  8.10
```

**→ Adresse 007962 in N2814 zeigt auf einen OPEN**

### → Gesamtergebnis:

- WAIT-Zeiten gehen auf die Dateiverarbeitung zurück
- Programm-technisch keine Optimierung möglich
- Buffer überprüfen gegen Empfehlungen von verantwortlichen Stellen (gibt es schon Empfehlungen??)

## DB2 – Index – 1

Date: 2002.11.14 Job: TN3A5K07 N3A56 IKJEFT01

----- JOB ENVIRONMENT -----			----- MEASUREMENT STATISTICS -----		
PROGRAM MEASURED	-	IKJEFT1B	CPS TIME PERCENT	-	95.21
JOB NAME	-	TN3A5K07	WAIT TIME PERCENT	-	4.79
JOB NUMBER	-	JOB22610	RUN MARGIN OF ERROR PCT	-	.94
STEP NAME	-	N3A56.IKJEFT01	CPU MARGIN OF ERROR PCT	-	.96
DATE OF SESSION	-	11/14/2002	TOTAL SAMPLES TAKEN	-	50,844
TIME OF SESSION	-	00:44:54	TOTAL SAMPLES PROCESSED	-	10,843
CONDITION CODE	-	C-0000	INITIAL SAMPLING RATE	-	11.11/SEC
			FINAL SAMPLING RATE	-	0.69/SEC
SYSTEM	-	z/OS 01.01.00	SESSION TIME	-	259 MIN 53.57 SEC
DFSMS	-	2.10.0	CPU TIME	-	202 MIN 1.73 SEC
SUBSYSTEM	-	DB2 7.1.0	WAIT TIME	-	10 MIN 9.37 SEC
DB2 SUBSYSTEM ID	-	D203	STRETCH TIME	-	47 MIN 42.46 SEC
DB2 APPLICATION	-	N3A56			

**→ CPU-Analyse erforderlich**



## DB2 – Index – 2

Achtung: Nur Beschreibung des DB2-Moduls DSNK2DM; dies hat nichts mit dem SQL Fetch zu tun!

#IEP

MODULE NAME	SECTION NAME	LINE NUMBER	PROCEDURE NAME	START LOC	% CPU TIME SOLO TOTAL
.DB2	DSNK2DM		DSNKFTCH		66.69 66.70
.DB2	DSNK2DM		DSNKNXT2		22.55 22.60
.DB2	DSNBBM		DSNB1GET		6.20 6.21
.DB2	DSNBBM		DSNB1REL		2.41 2.41
.DB2	DSNBBM		DSNB1CPF		.53 .53
.DB2	DSNVSR		SUSP/RES/CANCEL		.33 .33
.DB2	DSNXGRDS		RDS ACCESS MODULE		.30 .30
.DB2	DSNIDM		DATA MANAGEMENT		.28 .28
.NUCLEUS	IEAVESLK		SUSPEND LOCK SERVICE		.07 .07
N3A56				01BD00	.04 .04

→ schlechte Index-Nutzung

### → Gesamtergebnis

- Hohe %-Zahl bei dem Text “FETCH TYPE 2 IDX ...” weist auf schlechte Index-Nutzung hin
- Index fehlt oder wird nicht / kaum benutzt
- In Kapitel #SUS findet man den DBRM
- In Kapitel #CSS findet man den SQL
- auch ein fehlender Run-Stats kann die Ursache sein
- Kontaktaufnahme mit DBA

## DB2 – Table-Space-Scan – 1

Date: 2003.06.06 Job: MN5952Q1 N5952 IKJEFT01

→ CPU-Analyse war bei diesem Job erforderlich

#IEP

**Achtung:** Nur Beschreibung des DB2-Moduls DSNK2DM; dies hat nichts mit dem SQL Fetch zu tun!

MODULE NAME	SECTION NAME	LINE NUMBER	PROCEDURE NAME	START LOC	% CPU SOLO	TIME TOTAL
.DB2	DSNIDM		DSNIRNXT FETCH NEXT ROW TO PROG		68.53	68.53
.DB2	DSNK2DM		DSNKFTCH FETCH TYPE 2 IDX ENTRY		5.93	5.93
.DB2	DSNBBM		DSNB1GET RETRIEVE REQUESTED PAGE		4.49	4.49
.DB2	DSNXGRDS		DSNXSINE RETR/BLD BLK OF SRT RECS		3.60	3.60
.DB2	DSNXGRDS		DSNXSTSE RDS TREE SORT MODULE		2.73	2.73
.DB2	DSNXGRDS		DSNXSMRE RDS MERGE MOD		1.50	1.50
.DB2	DSNBBM		DSNB1REL PAGE RELEASE ROUTINE		1.40	1.40
.DB2	DSNXGRDS		DSNXRRP RTIME RESIDUAL PRED EXEC		1.27	1.27
.DB2	DSNIDM		DSNIONX2 NEXT ON CUB ON IXED FAN		1.02	1.02
.DB2	DSNK2DM		DSNKNXT2 FETCH TYPE 2 IDX ENTRY		1.01	1.01

**→ Hinweis auf Table-Space-Scan**  
**→ Wer ist Verursacher?**

## DB2 – Table-Space-Scan – 2

Chapter : #SUS

SQL	SQL	STMT-EXECUTION	TIME/CNT	% CPU	TIME	MARGIN OF ERROR	.77%	
TYPE	NAME	CNT	AVG TIME	SOLO	TOTAL	00	29.00	58.00
DBRM	N5B18	943	.4409	14:42:13	20.00	20.00	*****	
DBRM	N5J00	4,004	.0163	13:39:10	2.11	2.11	*	
DBRM	N5J08	1,501	.1341	10:37:59	10.27	10.27	****	
DBRM	N5J22	1,489	.0839	10:43:08	6.34	6.34	***	
DBRM	N5X78	1,501	.7503	07:59:40	56.97	56.97	*****	
TOTAL SQL CPUUSAGE				97.79	97.79			

→ es könnte 3-4 Verursacher geben, daher

→ Packages genauer prüfen

→ hier nicht aufgeführt, aber ...

→ Kapitel #ACE gibt Hinweise, wer der Aufrufer des häufig benutzten DB2-Moduls „FETCH NEXT ROW...” ist

→ hier Konzentration auf N5X78

## DB2 – Table-Space-Scan – 3

#CSS

DBRM - N5X78

CREATED - 10/17/2002 07:59:40

LOCATION: DECOM\_DB2N

STATIC, NON-CURSOR SQL

7078 DELETE FROM RWAEZUVI WHERE MANDANTK=:H AND KTOINHNR=:H AND RWKTONR=:H AND  
ISOWHRCD=:H

STMT NUMBER	STATEMENT TEXT	STMT-EXECUTION CNT	AVG-TIME	% CPU TIME SOLO	MARGIN OF ERROR TOTAL	00	29.00	.77% 58.00
7078	DELETE	1,501	.7503	56.97	56.97	*****		
	TOTALS	1,501	.7503	56.97	56.97			

→ Es ist der DELETE

### → Gesamtergebnis

- genau der betrachtete DELETE ist der Verursacher
- Kontaktaufnahme mit DBA erforderlich

## DB2 – Aufrufzahlen – 1

Date: 2003.06.28 Job: WN6402J2 N6402 IKJEFT01

#MSD

```
----- JOB ENVIRONMENT -----      ----- MEASUREMENT STATISTICS -----
PROGRAM MEASURED   -   IKJEFT1B      CPS TIME PERCENT   -   94.55
JOB NAME           -   WN6402J2      WAIT TIME PERCENT  -   5.45
JOB NUMBER         -   JOB18314      RUN MARGIN OF ERROR PCT -   .83
STEP NAME         -   N6402.IKJEFT01  CPU MARGIN OF ERROR PCT -   .85
DATE OF SESSION    -   06/28/2003     TOTAL SAMPLES TAKEN  -   24,044
TIME OF SESSION    -   02:32:27      TOTAL SAMPLES PROCESSED - 14,043
CONDITION CODE     -   C-0000        INITIAL SAMPLING RATE- 1.68/SEC
                                           FINAL SAMPLING RATE  - 0.84/SEC

SYSTEM -          z/OS   01.03.00
DFSMS           -          1.3.0   SESSION TIME - 278 MIN 8.60 SEC
SUBSYSTEM       -          DB2 7.1.0 CPU TIME - 216 MIN 27.68 SEC
DB2 SUBSYSTEM ID -          DB2N   WAIT TIME - 12 MIN 28.27 SEC
DB2 APPLICATION -          N6402   STRETCH TIME - 49 MIN 12.64 SEC
```

**→ sehr hoher CPU-Verbrauch im Vergleich zur WAIT-Zeit**  
**→ CPU-Analyse erforderlich**

## DB2 – Aufrufzahlen – 2

#IEP

MODULE	SECTION	LINE	PROCEDURE	START	% CPU TIME	
NAME	NAME	NUMBER	NAME	LOC	SOLO	TOTAL
.NUCLEUS	IEAVSTA1		COMM TASK ESTAE		6.93	6.93
.DB2	DSNK2DM		DSNKFTCH FETCH TYPE 2 IDX ENTRY		6.12	6.12
.DB2	DSNXGRDS		DSNXRTIM RDS ACCESS MODULE GENER		5.71	5.71
.DB2	DSNXGRDS		DSNXERD TOPMOST RDS CSECT		5.68	5.68
.DB2	DSNIDM		DSNISFX2 SET FUNC TYPE 2 IDX SCAN		4.87	4.87
.PRIVATE			PRIVATE AREA		4.47	4.47
.DB2	DSNIDM		DSNISRID SET CUB BY LST OF RIDS		3.80	3.80
.DB2	DSN3EPX		DSNAPRHX PGM REQUEST APPL INTERFC		3.50	3.50
.DB2	DSNBBM		DSNB1GET RETRIEVE REQUESTED PAGE		2.61	2.61
.DB2	DSNXGRDS		DSNXERT APPLICATION CALL ROUTINE		2.43	2.43

→ kein eindeutiger Verursacher außer DB2  
→ ist es (im) DB2?



## DB2 – Aufrufzahlen – 3

#PSU

MODULE	SECTION	16M	FUNCTION	% CPU TIME		MARGIN OF ERROR	.85%
NAME	NAME	<,>		SOLO	TOTAL	00	35.00 70.00
.SYSTEM	.COBLIB		COBOL LIBRARY SUBROUTI	.89	.89		
.SYSTEM	.DB2		DB2 SYSTEM SERVICES	69.63	69.63	*****	
.SYSTEM	.NUCLEUS		MVS NUCLEUS	10.25	10.25	***	
.SYSTEM	.PRIVATE		PRIVATE AREA	4.47	4.47	**	
.SYSTEM	.SMS		SYSTEM MANAGER STORAGE	.64	.64		
				-----	-----		
.SYSTEM	TOTALS		SYSTEM SERVICES	87.11	87.11		
N2X20	N2X20			2.09	2.09	*	
				-----	-----		
N2X20	TOTALS	>		2.11	2.11		
XXA08	XXA08	<		7.40	7.40	***	
ZFU23		<		1.48	1.48		
				-----	-----		
PROGRAM	IKJEFT1B	TOTALS		100.00	100.00		

→ Der Verbrauch liegt im DB2

## DB2 – Aufrufzahlen – 4

#SUS

SQL	SQL	STMT-EXECUTION	TIME/CNT	% CPU	TIME	MARGIN OF ERROR	.85%
TYPE	NAME	CNT	AVG-TIME	SOLO	TOTAL	00	28.00 56.00
DBRM	N1X85	141,498	.0001	07:48:23	2.08	2.08	*
DBRM	N2J90	126,784	.0000	13:39:43	1.17	1.17	
DBRM	N2X20	8,189,558	.0000	11:04:42	55.17	55.17	*****
DBRM	N6K90	403,801	.0003	06:54:01	10.46	10.46	****
TOTAL SQL CPUUSAGE				68.88	68.88		

**→ hohe Zahlen des Package machen die Last**

### → Gesamtergebnis

→ Topverbraucher ist das DBRM N2X20

→ die Aufrufzahlen sind zu plausibilisieren

## COBOL-Befehle – 1

Date: 2003.06.02 Job: TI9I9G08 I9G08 IMSBMPP

```
----- JOB ENVIRONMENT -----          ----- MEASUREMENT STATISTICS -----
PROGRAM MEASURED   -   DFSRRC00          CPS TIME PERCENT   -   85.70
JOB NAME           -   TI9I9G08          WAIT TIME PERCENT  -   14.30
JOB NUMBER        -   JOB26220          RUN MARGIN OF ERROR PCT -   .77
STEP NAME         -   I9G08.IMSBMPP      CPU MARGIN OF ERROR PCT -   .84
DATE OF SESSION   -   06/02/2003        TOTAL SAMPLES TAKEN -   16,064
TIME OF SESSION   -   17:10:37          TOTAL SAMPLES PROCESSED - 16,064
CONDITION CODE    -   C-0000            INITIAL SAMPLING RATE- 7.58/SEC
                                                    FINAL SAMPLING RATE  - 7.58/SEC

SYSTEM -          z/OS   01.03.00
DFSMS             -          1.3.0        SESSION TIME -      35 MIN 18.30 SEC
SUBSYSTEM        -   IMS BMP 6.1 L=S      CPU TIME -          24 MIN 39.71 SEC
                                                    WAIT TIME   -       4 MIN  6.88 SEC
                                                    STRETCH TIME -      6 MIN 31.70 SEC
DB2 SUBSYSTEM ID -          DB2Q
```

**→ CPU-Analyse erforderlich**

## COBOL-Befehle – 2

#IEP

MODULE	SECTION	LINE	PROCEDURE	START	% CPU TIME	
NAME	NAME	NUMBER	NAME	LOC	SOLO	TOTAL
.COBLIB	IGZCPAC		IGZCIN1 (V3) INSPECT		22.69	22.69
.COBLIB	IGZCPAC		IGZCUST UNSTRING		15.18	15.18
.DB2	DSNK2DM		DSNKFTCH FETCH TYPE 2 IDX ENTRY		4.10	4.10
I9G10	I9G10			013280	3.66	3.66
.DB2	DSNIDM		DSNIOST2 SET ON CUB DEF ON IX FAN		3.20	3.20
.COMMON	.COMMONX		EXTENDED COMMON AREA		2.85	2.85
.DB2	DSNIDM		DSNIONX2 NEXT ON CUB ON IXED FAN		2.66	2.66
I9G10	I9G10			030A40	2.27	2.27
I9G10	I9G10			030A00	1.78	1.78
.DB2	DSNBBM		DSNB1GET RETRIEVE REQUESTED PAGE		1.69	1.69

→ Verursacher ist klar

### → Weitere Anmerkungen

- das Kapitel #ACE zeigt genau die Adressen, wo die Befehle INSPECT und UNSTRING aufgerufen werden; die Adresse (Adressumgebung) muss in der Umwandlungsliste gesucht werden
- es muss darauf geachtet werden, dass die Umwandlungsliste zum Laufzeitpunkt passt

### → Gesamtergebnis

- Topverbraucher ist das Programm
- Es ist zu prüfen, ob ohne großen Aufwand die CPU-Last auf INSPECT / UNSTRING verringert werden kann

## COBOL-Code – 1

Date: 2003.06.03 Job: MDEDEB12 DEB12 IKJEFT01

#MSD

```
----- JOB ENVIRONMENT -----
PROGRAM MEASURED - IKJEFT1B
JOB NAME - MDEDEB12
JOB NUMBER - JOB30383
STEP NAME - DEB12.IKJEFT01
DATE OF SESSION - 06/03/2003
TIME OF SESSION - 01:00:12
CONDITION CODE - C-0000

SYSTEM - z/OS 01.03.00
DFSMS - 1.3.0
SUBSYSTEM - DB2 7.1.0
DB2 SUBSYSTEM ID - DB2N
DB2 APPLICATION - DEB12

----- MEASUREMENT STATISTICS -----
CPS TIME PERCENT - 90.27
WAIT TIME PERCENT - 9.73
RUN MARGIN OF ERROR PCT - .97
CPU MARGIN OF ERROR PCT - 1.03
TOTAL SAMPLES TAKEN - 20,109
TOTAL SAMPLES PROCESSED - 10,108
INITIAL SAMPLING RATE - 1.68/SEC
FINAL SAMPLING RATE - 0.84/SEC

SESSION TIME - 200 MIN 5.84 SEC
CPU TIME - 156 MIN 28.26 SEC
WAIT TIME - 16 MIN 52.49 SEC
STRETCH TIME - 26 MIN 45.08 SEC
```

→ CPU-Analyse folgt

## COBOL-Code – 2

#IEP

MODULE	SECTION	LINE	PROCEDURE	START	% CPU	TIME
NAME	NAME	NUMBER	NAME	LOC	SOLO	TOTAL
DEU64	DEU64			000EC0	23.48	23.48
DEU56	DEU56			0023C0	17.79	17.79
.NUCLEUS	IEAVSTA1		COMM TASK ESTAE		4.22	4.22
.DB2	DSNXGRDS		DSNXERD TOPMOST RDS CSECT		4.04	4.04
.PRIVATE			PRIVATE AREA		3.65	3.65
.DB2	DSNIDM		DSNIOST2 SET ON CUB DEF ON IX FAN		2.51	2.51
.DB2	DSN3EPX		DSNAPRHX PGM REQUEST APPL INTERFC		2.22	2.22
.DB2	DSNXGRDS		DSNXERT APPLICATION CALL ROUTINE		2.19	2.19
.DB2	DSNXGRDS		DSNXECP COPY APPLCTN STRUCTURES		1.56	1.56
.DB2	DSNK2DM		DSNKFTCH FETCH TYPE 2 IDX ENTRY		1.49	1.49

→ Verursacher ist klar



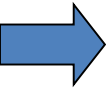
### → Weitere Anmerkungen

- die Adressen (Adressumgebung) müssen in den Umwandlungslisten gesucht werden
- es muss darauf geachtet werden, dass die Umwandlungsliste zum Laufzeitpunkt passt

### → Gesamtergebnis

- Es ist zu prüfen, ob ohne großen Aufwand die CPU-Last auf dem entsprechenden Code verringert werden kann.
- Hinweise geben die veröffentlichten Empfehlungen (sofern vorhanden).

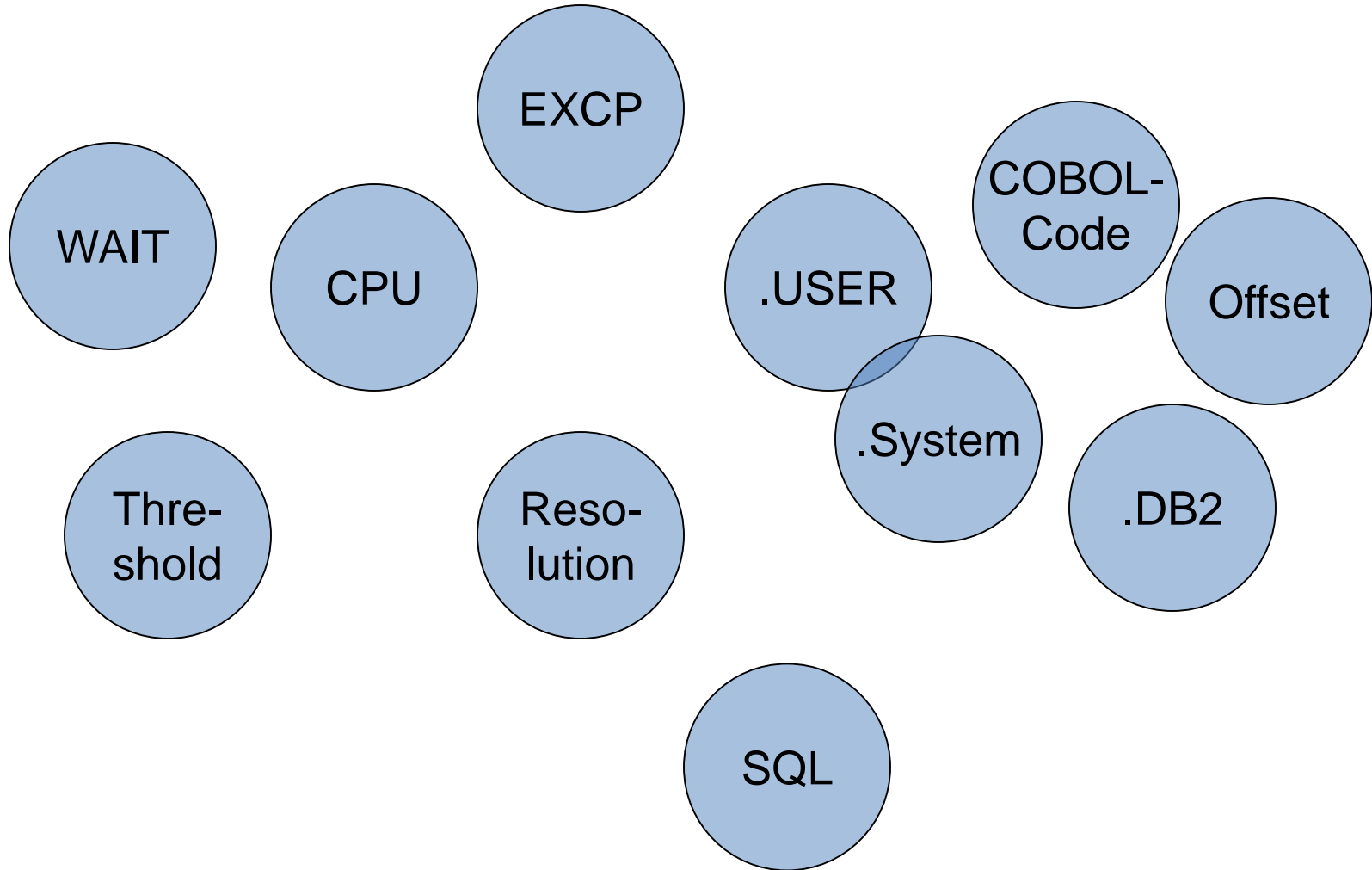


- 
- Vorstellung und Einführung
  - Optimierungen – Beispiele und Potential
  - Strobe – Handling
  - Vorgehensweise für die Analyse (Strobe)
  -  • Vorgehensweise für die Analyse (iStrobe)
  - Modellierung und DB2-Zugriffe
  - COBOL–Felder – COBOL-Befehle
  - Auswirkungen von Optionen – COBOL – LE
  - Praxisbeispiele – Diskussion – Austausch

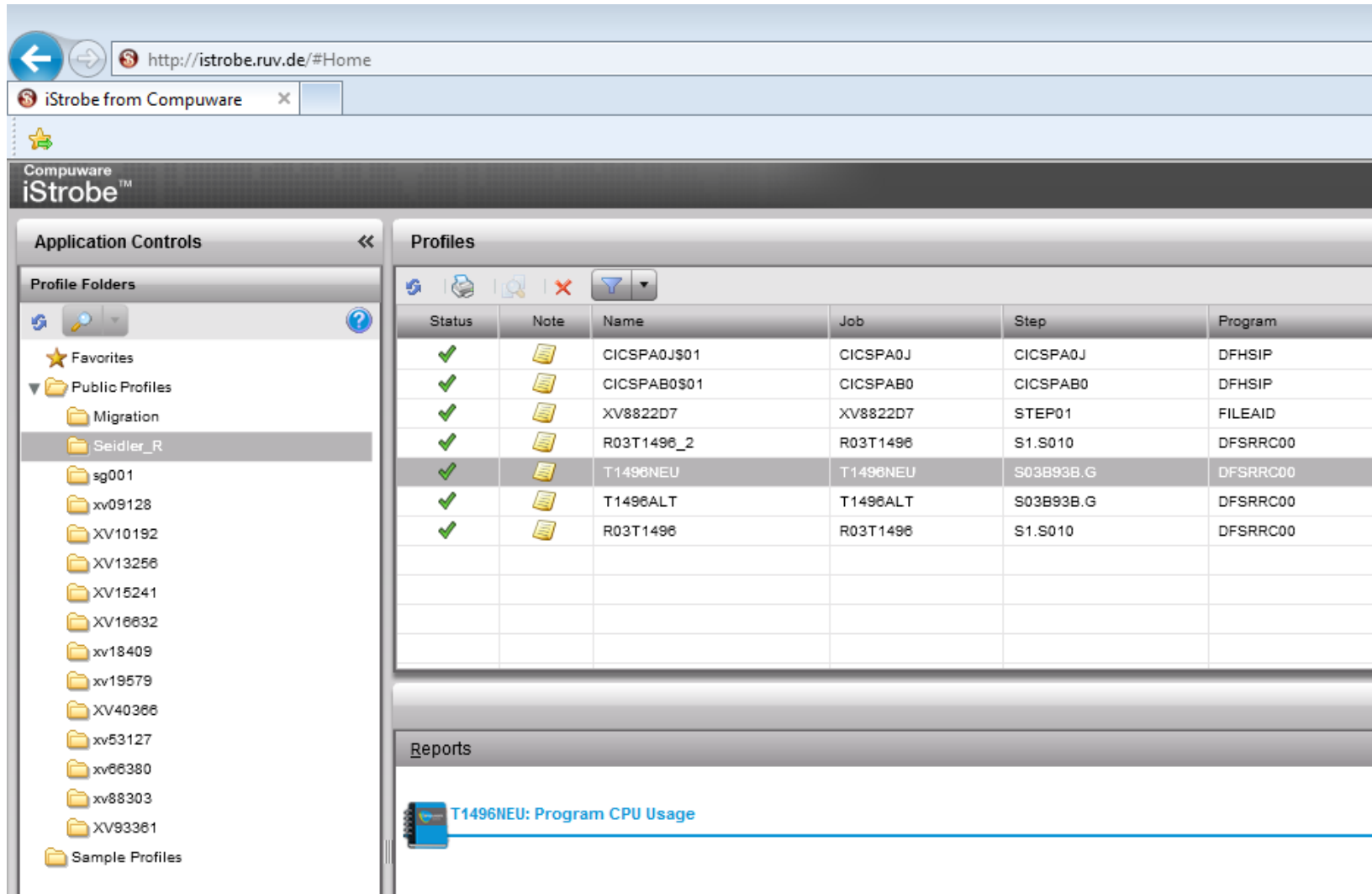
# Vorgehensweise für die Analyse (Strobe)

## Begriffe

---



## Directories – kein Schutz eingerichtet 😊










The screenshot shows the iStrobe web interface. The browser address bar displays <http://istrobe.ruv.de/#Home>. The page title is "Compuware iStrobe". The interface is divided into two main sections: "Application Controls" on the left and "Profiles" on the right.

**Application Controls**


**Profile Folders**

- ★ Favorites
- Public Profiles
  - Migration
  - Seidler\_R
  - sg001
  - xv09128
  - XV10192
  - XV13256
  - XV15241
  - XV16632
  - xv18409
  - xv19579
  - XV40366
  - xv53127
  - xv66380
  - xv88303
  - XV93361
  - Sample Profiles

**Profiles**

Status	Note	Name	Job	Step	Program
✓		CICSPA0J\$01	CICSPA0J	CICSPA0J	DFHSIP
✓		CICSPAB0\$01	CICSPAB0	CICSPAB0	DFHSIP
✓		XV8822D7	XV8822D7	STEP01	FILEAID
✓		R03T1496_2	R03T1496	S1.S010	DFSRRCC00
✓		T1496NEU	T1496NEU	S03B93B.G	DFSRRCC00
✓		T1496ALT	T1496ALT	S03B93B.G	DFSRRCC00
✓		R03T1496	R03T1496	S1.S010	DFSRRCC00

**Reports**

 [T1496NEU: Program CPU Usage](#)

# Analyse mit iStrobe

## Überblick

Compuware  
**iStrobe™** Logged in as iStrobe [iStrobe Help](#)

Application Controls

Reports

T1496NEU: Measurement Session Data

Job environment	
Job	T1496NEU
Job number	JOB06822
Address space ID	016B
Step	S03B93B.G
Program	DFSRR00
Session start date	2013-10-07
Session start time	11:59:50
Condition code	C0000
System	z/OS 01.13.00
Data facility	DFSMS 1.13.0
CPU model	2817-712
SMF/System ID	MVSB/MVSB
LPAR	MVSB
64-bit architecture	Enabled
Number of zIIPs	1
Region size < 16M	8,256K
Region size > 16M	263,168K
Measurement release	04.03.00
Reporter release	04.03.00
Strobe PTF level	002652/M02665/043007

Percent time	
Execution	73.60 %
Wait	26.40 %
Runtime margin of error	0.58 %
CPU margin of error	0.68 %

Processor time	
CPU	10 min 04.09 sec
SRB time	0.34 sec
Service units	33,677,413

Clock time	
Session duration	17 min 12.47 sec
Execution	10 min 04.09 sec
Wait	3 min 36.68 sec
Stretch	3 min 31.69 sec
Multiprocessor ratio	1.00

Subsystems	
DB2 9.1.0 NFM	
IMS BMP 11.1 L=S	
IMS module	IMS P03B93BT
DB2 sysid	DBT1

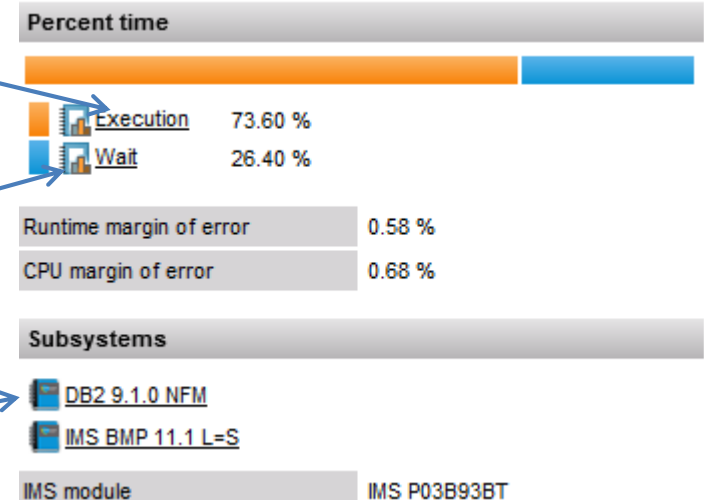
Measurement parameters	
Est. session time	10 min
Target sample size	20,000
Request number	970
Request type	Queued
Request start date	
Request start time	
Final action	
Profile run date	2013-10-07
Profile run time	12:35:04

Measurement statistics	
Total samples taken	28,181
Samples processed	28,181
Initial sampling rate	33.33/sec
Final sampling rate	33.33/sec
EXCP count	20,061
EXCP rate	19.43/sec
Pages in	0
Pages out	0
Page rate/sec	0.00

Sample data set USER.STROBE.T1496NEU.S002D001.STROBE

## Prinzipien









- unterstrichene Links
  - Info über Punkt *oder*
  - Link zu Detaillierung
- Bild mit Balken
  - Link zu Top-Info
- Bild mit Buch
  - Link zu detaillierten Infos
  - Ebene tiefer / wie aufklappen
- Immer auf mouseover achten
- nächstes Release wird anders!!! ☹️ 😊













# Analyse mit iStrobe

RESOLUTION ==> 0004 vs. RESOLUTION ==> 0040 (hex)

Name	Job	Step	Program	Start Date	Start Time
P98U20501	XV52913P	NEU	P98N721	2014-02-03	07:12:14
P98U20502	XV52913P	NEU	P98N721	2014-02-03	07:12:14

Module	Section	Starting location	Type	CPU %
P98U205	P98U205	 <a href="#">0000362C</a>	USER	6.83
		 <a href="#">000035A8</a>		5.83
		 <a href="#">0000359C</a>		5.21
		 <a href="#">000032C4</a>		3.97
		 <a href="#">000032E4</a>		3.97
		 <a href="#">000032D8</a>		3.85
		 <a href="#">000035B4</a>		3.22
		 <a href="#">000045E4</a>		3.10

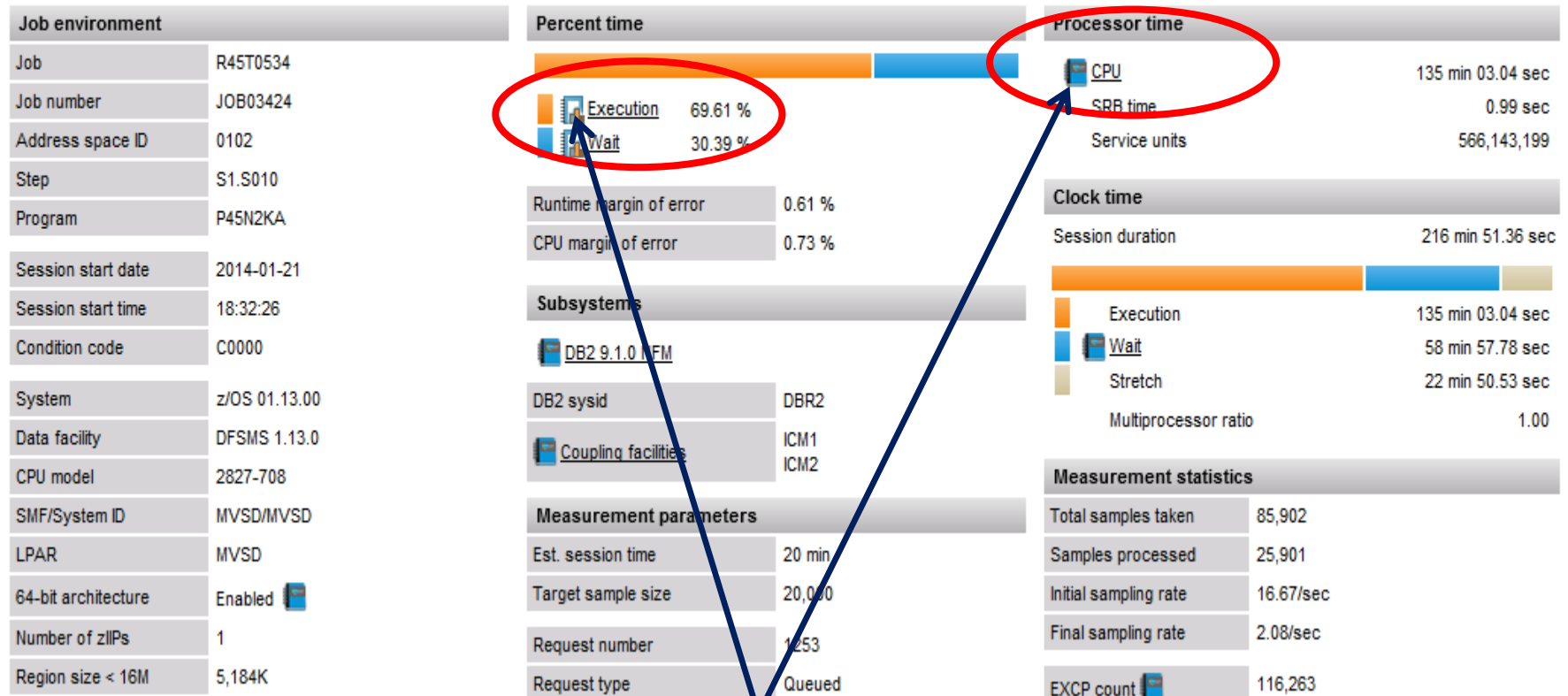
Module	Section	Starting location	Type	CPU %
P98U205	P98U205	 <a href="#">00003580</a>	USER	20.99
		 <a href="#">000032C0</a>		14.03
		 <a href="#">000045C0</a>		11.18
		 <a href="#">00003600</a>		9.93
		 <a href="#">00004580</a>		6.83
		 <a href="#">00003280</a>		4.22
		 <a href="#">00004600</a>		3.35
		 <a href="#">00004540</a>		2.85
		 <a href="#">00003180</a>		1.98
		 <a href="#">00002C40</a>		1.36
IGZCPAC			SYSTEM	0.99

Beginn des Bereichs

# CPU-Analyse

## DB2 – Index – 1

R45T0534: Measurement Session Data



→ CPU-Analyse



## DB2 – Index – 2













R45T0534: Top CPU Resource Consumers Threshold: 0.1

Top Wait Resource Consumers    Top EXCP Resource Consumers

Transaction	Pseudo-section	Module	Section	Starting location	Type	CPU %
P45N2KAR	.DB2	DSNK2DM	DSNKFTCH (18.32 %)		SYSTEM	18.32
Query	Timestamp	Exec stmt number	Exec stmt text	Invoking stmt	Type	CPU %
P45NRDR	01 10 2014 02:54:31 PM	505	OPEN	7	USER	12.53
Pseudo-section	Module	Section	Starting location	Type	CPU %	
P42HH52	P42HH52		000027C0	USER	11.59	
			00007A80		10.17	
			00000980		7.40	
Query	Timestamp	Exec stmt number	Exec stmt text	Invoking stmt	Type	CPU %
P45NRAU	09 10 2013 01:50:35 PM	2300	FETCH	6	USER	11.12
Transaction	Pseudo-section	Module	Section	Starting location	Type	CPU %
P45N2KAR	.COBLIB	IGZCPAC	IGZCXDI (5.05 %)		SYSTEM	5.05
P45N2KAR	.DB2	DSNK2DM	DSNKNXT2 (4.11 %)		SYSTEM	4.11
Pseudo-section	Module	Section	Starting location	Type	CPU %	
P42H256	P42H256		000074...	USER	3.91	
Query	Timestamp	Exec stmt number	Exec stmt text	Invoking stmt	Type	CPU %
P45NRAU	09 10 2013 01:50:35 PM	2160	FETCH	1	USER	3.76

→ näher schauen

## DB2 – Index – 3

				CPU %			
				Solo	Total		
		Totals		99.58	100.00		
Pseudo-module 	Description	Attrib	Comb	Solo	Total		
 .SYSTEM	SYSTEM SERVICES	0.00	44.06	43.67	44.06		
Pseudo-section 	Description	Attrib	Comb	Solo	Total		
  .COBLIB	COBOL LIBRARY SUBROUTINE	0.00	9.60	9.60	9.60		
 .COMMON	COMMON AREA	0.00	0.58	0.58	0.58		
  .DB2	DB2 SYSTEM SERVICES	0.00	30.25	29.90	30.25		
Module 	RMODE	Description	Interval length	Attrib	Comb	Solo	Total
DSNALI		LANGUAGE INTERFACE	1040	0.00	0.31	0.31	0.31
DSNBBM		BM BUFFER POOL MGR	1486848	0.00	1.14	1.13	1.14
 DSNIDM		DATA MANAGEMENT DRIVER	52136	0.00	2.15	2.05	2.15
 DSNK2DM		IDX MGR	103296	0.00	22.55	22.42	22.55
Section 	Language	Description		Attrib	Comb	Solo	Total
DSNKFTCH		FETCH TYPE 2 IDX ENTRY		0.00	18.32	18.20	18.32
DSNKNXT2		FETCH NEXT TYPE 2 INDEX		0.00	4.12	4.10	4.12
DSNKTRAV		SEARCH INDEX FOR KEY		0.00	0.11	0.10	0.11
Module	RMODE	Description	Interval length	Attrib	Comb	Solo	Total
DSNWWAPR		APPEND DATA AND ROUTE	73264	0.00	0.16	0.16	0.16

→ Index-Nutzung

## DB2 – Index – 4

145T0534: Attribution of CPU Usage

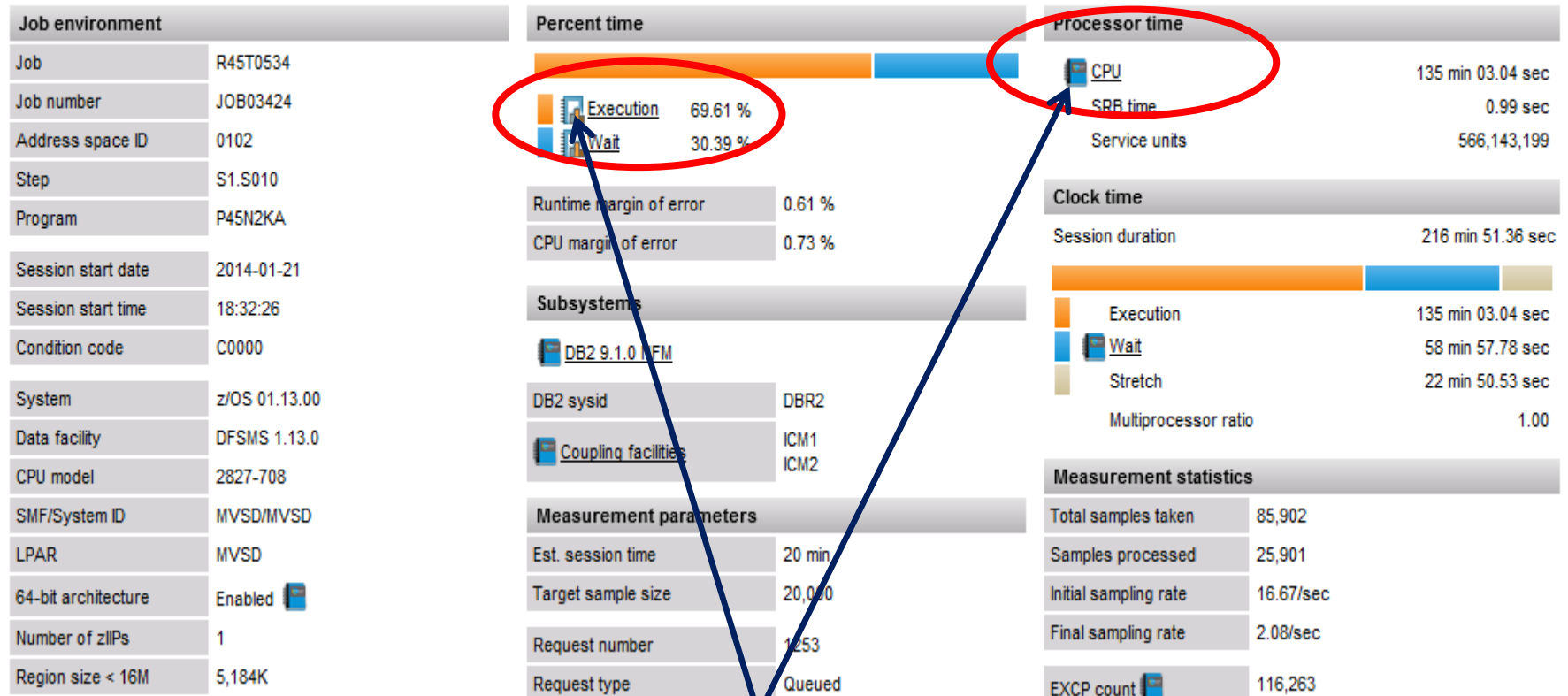
Threshold: 0.1

								CPU %	
								Solo	Total
						Totals		27.36	27.49
Pseudo-module	Module	Section	Description		Solo	Total			
COBLIB	IGZCPAC	IGZCXDI	DOUBLE PRECISION DIVIDE		5.05	5.05			
DB2	DSNK2DM	DSNKFTCH	FETCH TYPE 2 IDX ENTRY		18.20	18.32			
		SQL	Executing	Target					
Transaction	Stmt	Query	Time	Stmt	Text	Stmt	Text	Solo	Total
P45N2KAR	SQLQ	P45NRAU	13.50.35	2300	FETCH	06	DECLARE	10.53	10.62
P45N2KAR	SQLQ	P45NRDR	14.54.31	505	OPEN	07	DECLARE	7.42	7.44
Pseudo-module	Module	Section	Description		Solo	Total			
DB2	DSNK2DM	DSNKNXT2	FETCH NEXT TYPE 2 INDEX		4.10	4.12			
		SQL	Executing	Target					
Transaction	Stmt	Query	Time	Stmt	Text	Stmt	Text	Solo	Total
P45N2KAR	SQLQ	P45NRDR	14.54.31	505	OPEN	07	DECLARE	4.03	4.03

→ Index-Nutzung prüfen (lassen); explain

## IGZ-Routine – 1

R45T0534: Measurement Session Data

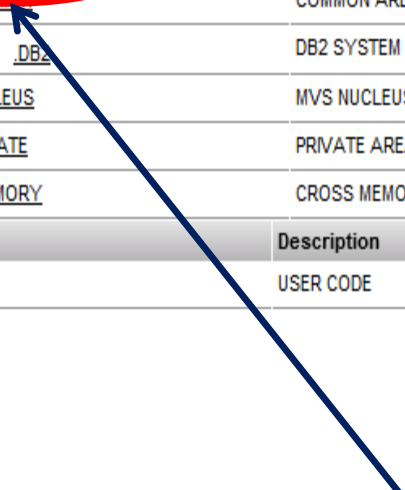
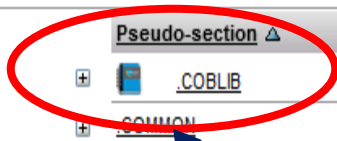


→ CPU-Analyse

## IGZ-Routine – 2

R45T0534: Program CPU Usage Threshold: 0.1





		CPU %			
				Solo	Total
Totals				99.58	100.00
Pseudo-module	Description	Attrib	Comb	Solo	Total
[-] .SYSTEM	SYSTEM SERVICES	0.00	44.06	43.67	44.06
Pseudo-section					
[+] .COBLIB	COBOL LIBRARY SUBROUTINE	0.00	9.60	9.60	9.60
[+] .COMMON	COMMON AREA	0.00	0.58	0.58	0.58
[+] .DB2	DB2 SYSTEM SERVICES	0.00	30.25	29.90	30.25
[+] .NUCLEUS	MVS NUCLEUS	0.00	0.23	0.23	0.23
[+] .PRIVATE	PRIVATE AREA	0.00	0.41	0.41	0.41
[+] .XMEMORY	CROSS MEMORY	0.00	2.90	2.86	2.90
Pseudo-module	Description	Attrib	Comb	Solo	Total
[+] .USER	USER CODE	8.01	63.95	55.91	55.94



**→ Auslöser?**

## IGZ-Routine – 3

R45T0534: Attribution of CPU Usage Threshold: 0.1

				CPU %	
				Solo	Total
<b>Totals</b>				27.36	27.49
Pseudo-module 	Module	Section	Description	Solo	Total
 <b>.COBLIB</b>	<u>IGZCPAC</u>	<u>IGZCXDI</u>	DOUBLE PRECISION DIVIDE	5.05	5.05
Pseudo-module	Module	Section	Description	Solo	Total
 <u>.DB2</u>	<u>DSNK2DM</u>	<u>DSNKFTCH</u>	FETCH TYPE 2 IDX ENTRY	18.20	18.32
 <u>.DB2</u>	<u>DSNK2DM</u>	<u>DSNKNXT2</u>	FETCH NEXT TYPE 2 INDEX	4.10	4.12

**→ Klick auf – (ist ein +!!)**

## IGZ-Routine – 4

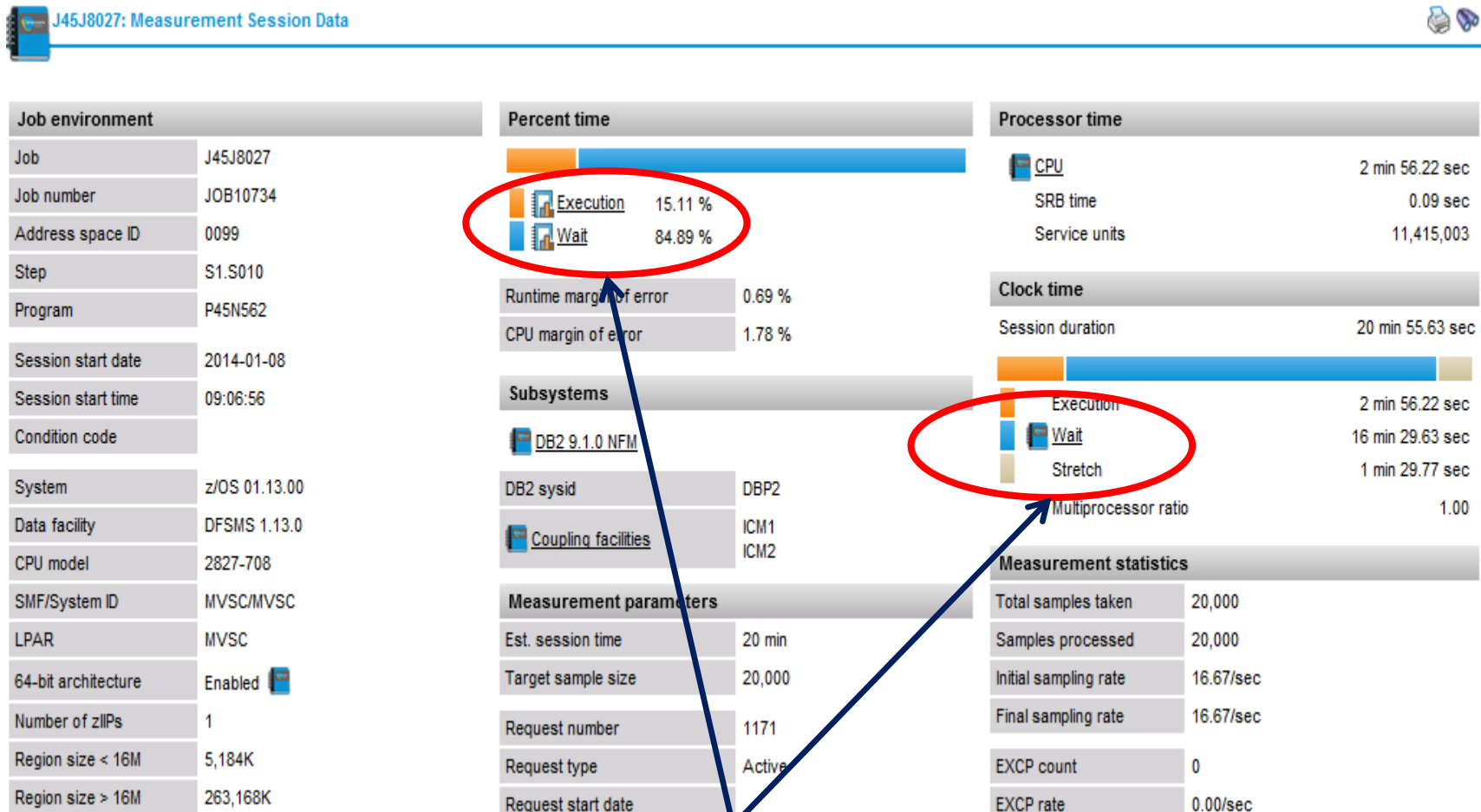
R45T0534: Attribution of CPU Usage Threshold: 0.1

								CPU %		
								Solo	Total	
<b>Totals</b>								27.36	27.49	
Pseudo-module $\Delta$	Module	Section	Description					Solo	Total	
$\square$ <u>COBLIB</u>	<u>IGZCPAC</u>	<u>IGZCXDI</u>	DOUBLE PRECISION DIVIDE					5.05	5.05	
Was invoked by						Via				
Transaction $\Delta$	Module	Section	Location	Line and source text / Description			Solo	Total		
P45N2KAR	P42HF13		00000CBC				0.29	0.29		
P45N2KAR	P42HF13		00000E90				0.28	0.28		
P45N2KAR	P42HF13		00001544				0.73	0.73		
P45N2KAR	P42HF13		00001666				0.12	0.12		
P45N2KAR	P42H029		000032E4				0.14	0.14		
P45N2KAR	P42H029		00003478				0.13	0.13		
P45N2KAR	P42H115		00012B2C				0.26	0.26		
P45N2KAR	P42H115		00012BE6	<u>IGZCPAC</u>	<u>IGZCXPR</u>	DOUBLE PRECIS. EXPONENT.	62	2.62		
P45N2KAR	P42H115		00012C6A				0.17	0.17		
$\oplus$ <u>_DB2</u>	<u>DSNK2DM</u>	<u>DSNK2FCH</u>	FETCH TYPE 2 IDX ENTRY					18.20	18.32	

**→ Offset in Compileliste aufsuchen**

# WAIT-Analyse

## DB2 – 1



**→ WAIT-Analyse**



# WAIT-Analyse

## DB2 – 2

J45J8027: Top Wait Resource Consumers Threshold:

[Top CPU Resource Consumers](#) [Top EXCP Resource Consumers](#)

	Pseudo-section	Module	Section	Starting location	Type	Wait %	
	<u>.NUCLEUS</u>	IEAVEPS1			SYSTEM	84.88	
	Query	Timestamp	Exec stmt number	Exec stmt text	Invoking stmt	Type	Wait %
	P01NZ12	01 12 2006 10:23:24 AM	654	OPEN	1	USER	74.84
	P1NQ001	08 26 2013 09:23:21 AM	901	SELECT		USER	3.27
	P01NZ12	05 02 2011 09:51:29 AM	708	OPEN	15	USER	2.89
	P01NL16	06 17 2013 11:15:53 AM	1742	FETCH	17	USER	0.65
	P45NR32	03 08 2013 11:51:32 AM	1388	FETCH	13	USER	0.60
	P01NQ07	08 21 2007 07:27:29 AM	550	FETCH	16	USER	0.60
	P01NQ04	02 17 1999 04:53:12 PM	326	SELECT		USER	0.34
	P45NRCH	03 08 2013 11:50:34 AM	1454	FETCH	9	USER	0.25
	P45NR24	03 08 2013 11:50:21 AM	1195	FETCH	11	USER	0.22
	P45NR10	03 08 2013 11:49:29 AM	1659	FETCH	3	USER	0.17

**→ Jep. . . bitte mehr Details!**

# WAIT-Analyse

## DB2 – 3

J45J8027: DB2 Activity by Query

Threshold: 0.1

										Parallel Task CPU Distribution (sec)		Overall time (sec)		Wait %		
										SP CPU	Total CPU	Elapsed time	Total CPU	CPU %	Page	Total
<b>Totals</b>										0.000000	0.000000	1,239.997776	165.508364	93.87	0.00	84.76
Query name	Type	Count	Timestamp	Statement count	Avg time (sec)	Avg CPU time (sec)	Statement count	Avg time (sec)	SP CPU	Total CPU	Elapsed time	Total CPU	CPU %	Page	Total	
P01NZ12	DBRM		01 12 2006 10:23:24 AM	5,915	.187000	0.025798	5,460	.202581	0.000000	0.000000	1,106.107578	152.595656	86.54	0.00	74.97	
Executing stmt	Invoking stmt	Stmt type	Statement count	Avg time (sec)	Avg CPU time (sec)	SP CPU	Total CPU	Elapsed time	Total CPU	CPU %	Page	Total				
654	OPEN	1 DECLARE SC	455	2.430906	0.335303	0.000000	0.000000	1,106.062036	152.562808	86.53	0.00	74.84				
669	FETCH	1 DECLARE SC	5,005	.000006	0.000004	0.000000	0.000000	0.032056	0.022086	0.01	0.00	0.12				
P45NR32	DBRM		03 08 2013 11:51:32 AM	773,106	.000020	0.000008	257,778	.000045	0.000000	0.000000	15.457254	6.048539	3.43	0.00	0.60	
P45NR15	DBRM		03 08 2013 11:49:51 AM	150,444	.000018	0.000007	50,148	.000040	0.000000	0.000000	2.676679	0.979687	0.55	0.00	0.12	

→ Jep. . . und welcher SQL ist das?

# WAIT-Analyse

## DB2 – 4

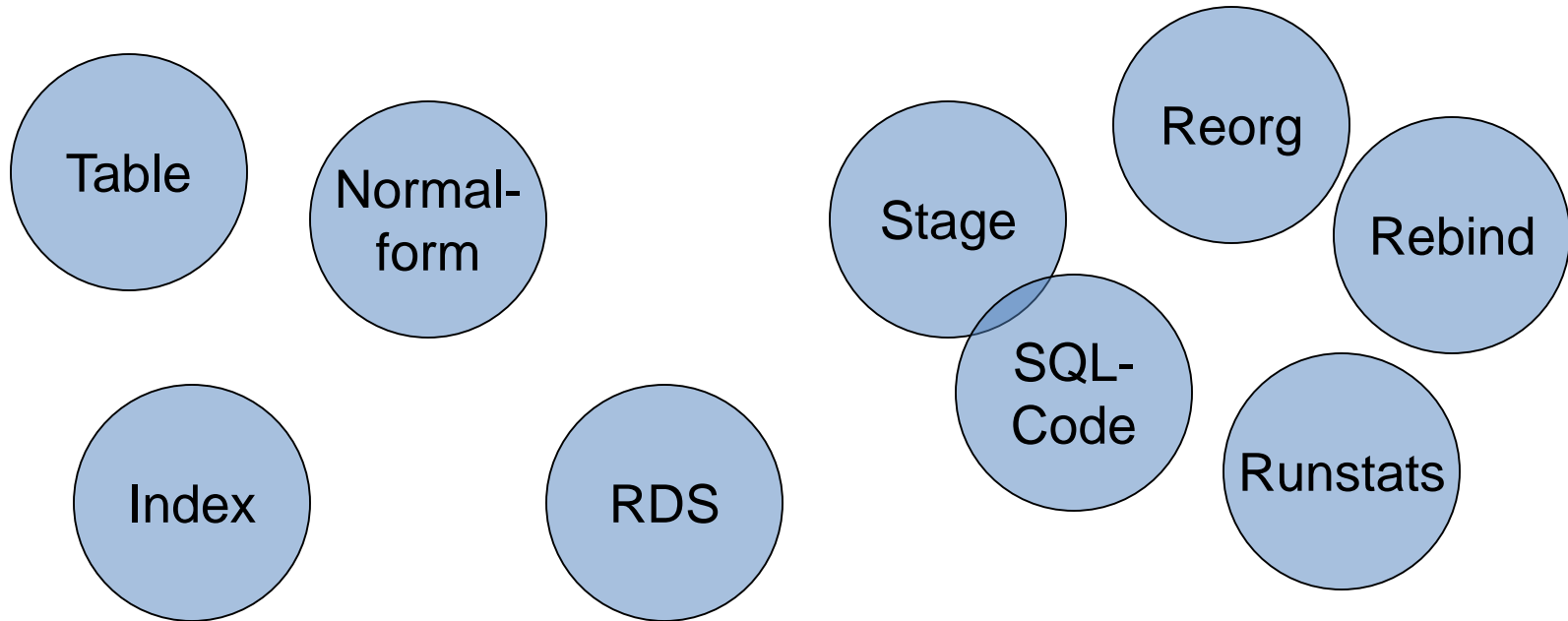
Query name	Type	Count	Timestamp	Statement count	time (sec)	Avg CPU time (sec)	Statement count	time (sec)	SP CPU	Total CPU	Elapsed time	Total CPU	CPU % ▾	Page
[-] P01NZ12	DBRM		01 12 2006 10:23:24 AM	5,915	.187000	0.025798	5,460	.202581	0.000000	0.000000	1,106.107578	152.595656	86.54	0.00
Executing stmt	Invoking stmt	Stmt type	Statement count	Avg time (sec)	Avg CPU time (sec)	SP CPU	Total CPU	Elapsed time	Total CPU	CPU % ▾	Page			
[-] 6A	OPEN	1 DECLARE	SC	455	2.430906	0.335303	0.000000	0.000000	1,106.062036	152.562808	86.53	0.00		
[-] SQL statement text											SQL 🔍			
[-] Run time statistics for scans executed by the SQL statement														
Type	Rows Processed	Rows Examined	Rows-Q Stage1	Rows-Q Stage2	Rows Inserted	Rows Deleted	Rows Updated	Get Page Requests	RI Page Scans	RI Page Deletes	LOB Page Scans	LOB Page Updates		
INDX	1,027,402	28,756,960	58,899,690	13,560,299	0	0	0	2,567,824	0	0	0	0		
SEQD	86,583,248	86,583,248	0	0	0	0	0	7,640,366	0	0	0	0		
SEQW	0	0	0	0	13,506,390	0	0	58,382	0	0	0	0		
[-] Services invoked														
[-] Parallel processing														
Distributed processing														
[-] 6B	FETCH	1 DECLARE	SC	5,005	.000006	0.000004	0.000000	0.000000	0.032056	0.022086	0.01	0.00		
[-] P45NR32	DBRM		03 08 2013	773,106	.000020	0.000008	257,702	.000045	0.000000	0.000000	15.457254	6.048539	3.43	0.00

→ ... der SQL und 'ne Menge mehr

- 
- Vorstellung und Einführung
  - Optimierungen – Beispiele und Potential
  - Strobe – Handling
  - Vorgehensweise für die Analyse (Strobe)
  - Vorgehensweise für die Analyse (iStrobe)
  - ➔ • Modellierung und DB2-Zugriffe
  - COBOL–Felder – COBOL-Befehle
  - Auswirkungen von Optionen – COBOL – LE
  - Praxisbeispiele – Diskussion – Austausch

## Begriffe

---



## Ziel von DB2 und SQL

---

- kodieren des WAS nicht des WIE 😊
- Aber:  
Modellierung<sup>(1)</sup>, Wartung<sup>(1,2,3)</sup> und Zugriff<sup>(2)</sup> haben großen Einfluss auf das WIE. ☹️
- Umgang damit – KISS ist (lebens)notwendig

<sup>(1)</sup> Datenmodell, Aufbau Tabellen, Aufbau Indizes

<sup>(2)</sup> SQL

<sup>(3)</sup> Änderung von Datenmengen, Art der Daten, Art der Abfragen ...

- Modellierung der Tabellen
- passende Nutzung der Runstats
- geeignete Nutzung von Reorgs
- angemessene Nutzung der Indexe
- richtiges Kodieren der SQLs



- Um einfache Relationen zu erhalten, wurde formalisierter Zerlegungsprozess für die Daten entwickelt.
- Es werden verschiedene Stufen für die Abhängigkeit der Daten untereinander definiert:
  - 1. Normalform
  - 2. Normalform
  - 3. Normalform
  - 4. Normalform
  - 5. Normalform



## Normalisierung – wie wäre es mit ...

---

- ...  
every entity depends  
  
on the key,  
  
the whole key,  
  
and nothing but the key

Formulierung: Danke an Gerhard Heiß.



## Normalisierung – Fragen

---

- Ist das denn noch normal?
- Das kann doch keiner mehr verstehen!
- Ist der ganze Quatsch denn notwendig?

- Normalisierungsprozess
  - ist aufwändig
  - liefert die Basis für stabile Datenstrukturen
  - Daten in 1. NF sind nicht sinnvoll verwaltbar
  - Daten in 2. NF sind schwierig verwaltbar
  - (mindestens) bis 3. NF durchführen
  - 5. NF „garantiert“ stabile Ergebnisse zur Laufzeit
- Denormalisierung für Physik immer möglich!

## Runstats

---

- Statistik zu einer Tabelle
- Beispiel:
- Anzahl der Zeilen
  - letzter Runstats
  - Anzahl pages
  - Anzahl indexpages
  - etc.
  - also alles, was ein Optimizer für seinen Zugriff braucht.

## Reorganisation einer Tabelle

---

- Reorg heißt u.a.
  - Neuaufbau der Tabelle
  - Neuaufbau des Index (Clustering)
- Ziel (denke an VSAM ;- )
  - leere Bereiche füllen
  - Überlaufbereiche neu anlegen
  - etc.
- Folgerung:  
regelmäßig Reorg durchführen  
... spätestens wenn Clusterratio <95%



## Indexdesign

---

- Zugriff muss durch Index unterstützt werden
  - Ausnahme: Minitabellen
- Ergebnis:
  - Tablespacescan wird vermieden
  - Non-matching Indexscan wird vermieden
  - oft werden interne Sorts nicht mehr benötigt
    - ascending / descending – ab V8 automatisch
- wichtigsten Index clustern
  - also nicht immer den primary index!

## allgemein

---

- Es lohnt sich, von Zeit zu Zeit einen Blick auf die Anwendung und die zugehörigen Tabellen zu werfen.
- Frage: Passt das Design der Tabelle zur Implementierung der Anwendung?
- Frage: Hat sich das Verhalten (Zugriffsarten) der Anwendung gegenüber “damals” verändert?



## DB2-Internas in aller Kürze

---

- Es gibt 5 Ebenen für die Zugriffe:
  - Stage 1
    - Data Manager mit einfachen Predicates
    - Indexmanager mit “matching index scan”
  - Stage 2
    - Daten laufen über RDS (Relational Data System)
  - Stage 3
    - virtuelle Predicates / set current timestamp
  - Stage 4
    - Alles andere, das bisher nicht abgedeckt ist
    - wie substr, timestamp auf Tabelle





- Matchcols möglichst hoch
  - Beispiel Telefonbuch
    - Nachname bekannt: Matchcol=1
    - Vorname zusätzlich bekannt: Matchcol=2
    - Straße zusätzlich bekannt: Matchcol=3
    - Hausnummer zusätzlich bekannt: Matchcol=4
- Matchcol=0 (non-matching indexscan) so schlecht wie Tablespacescan
  - Beispiel: Telefonbuch nach Straßennamen sortiert

# Modellierung und DB2-Zugriffe

## DB2-Indexdesign – Beispiel Versicherung 2008 – Ausgangspunkt

### UXU4490: DB2 Activity by Query

Query name	Type	Count	Timestamp	Statement count	CPU Avg time (sec)	Statement count	Wait Avg time (sec)	CPU %		Wait %	
								Solo	Total	Page	Total
Totals								82.47	82.47	0.00	0.85
U44663	DBRM		09 12 07 11:04:21 AM	180'398	.0012	79'795	.0027	82.47	82.47	0.00	0.85
Executing stmt		Invoking stmt		Stmt type	Statement count	Avg time (sec)	Solo	Total ↓	Page	Total	
1635 SELECT				STATIC NON-CURSOR	6'709	.0314	79.55	79.55	0.00	0.58	
SQL statement text								SQL	Predicate		
<pre> SELECT   TYP_WERT_DATUM INTO   :H :H FROM   UFLO_UTEXAZU WHERE   FK_POL_NR=:H   AND FK_OFFERT_VAR=0   AND GUELTIG_AB&lt;=:H   AND GUELTIG_BIS&gt;=:H   AND GUELTIG_AB&lt;&gt;GUELTIG_BIS   AND DN_ENTITY_ID_ATTTY=2100 ORDER BY   GUELTIG_AB DESC,   VERSION DESC FETCH FIRST 1 ROW ONLY WITH UR           </pre>											
Services invoked											
Pseudo-module	Module	Section	Description	Transaction	Solo	Total ↓	Page	Total			

## DB2-Indexdesign – Beispiel Versicherung 2008 – 1

Auszug direct aus DB2-Admin-Tool (Plan-Table):

```
-- SQL statement in PACKAGE : CONLINE.U44663.()
-- SQL in stmt: 1635
SELECT TYP_WERT_DATUM
INTO   :H :H
FROM   UFLO_UTEXAZU
WHERE  FK_POL_NR = :H
AND    FK_OFFERT_VAR = 0
AND    GUELTIG_AB <= :H
AND    GUELTIG_BIS >= :H
AND    GUELTIG_AB <> GUELTIG_BIS
AND    DN_ENTITY_ID_ATTTY = 2100
ORDER BY    GUELTIG_AB DESC
            , VERSION    DESC
FETCH FIRST 1 ROW ONLY
WITH UR
```

## DB2-Indexdesign – Beispiel Versicherung 2008 – 2

```
DB2 Admin ----- Interpretation of Row from $D2P0021.PLAN_TABLE ----- 17:18
```

```
Command ==>
```

```
Top of data
```

```
More: +
```

```
DB2 System: P0DB
```

```
Data as produced by EXPLAIN:
```

```
PLAN_TABLE Owner: $D2P0021
```

```
_ SELECT TYP_WERT_DATUM INTO :H :H FROM UFLO_UTEXAZU WHERE FK_POL_NR = :H AND F
_OFFERT_VAR = 0 AND GUELTIG_AB <= :H AND GUELTIG_BIS >= :H AND GUELTIG_AB <> GU
```

```
-----
! Matching index scan with scan of referenced data pages !
```

```
! PREFETCH through a page list will be performed. !
```

```
! !
```

```
. . .
```

```
-----
! Additional Sort for ORDER BY !
```

```
! !
```

```
. . .
```

## DB2-Indexdesign – Beispiel Versicherung 2008 – 3

```

                Index                Table                C C C
Select Index Name      Owner      Table Name      Owner      U   Cols G D L T
          *              *          *              *          *   * * * * *
-----
          UIEXAZU1      UFLO      UTEXAZU      UFLO      P    3 Y Y N 2
          UIEXAZU2      UFLO      UTEXAZU      UFLO      D    2 N N N 2
C          UIEXAZU3      UFLO      UTEXAZU      UFLO      D    4 N N N 2
***** END OF DB2 DATA *****

```

```

Sel Column Name      Seq No O Col Type Length Scale Null Def FP   Col Card
  *                * * *          *      * *   * *   *
-----
          FK_POL_NR      1 A INTEGER      4      0 N   N   N    17226
          FK_OFFERT_VAR  2 A SMALLINT     2      0 N   N   N     61
DI          FK_EX_ATTRIBUT_ID  3 D INTEGER      4      0 N   N   N    102
          FK_GUELTIG_AB  4 D DATE         4      0 N   N   N     2

```

# Modellierung und DB2-Zugriffe

## DB2-Indexdesign – Beispiel Versicherung 2008 – 4

```
DB2 Admin ----- PODB Column Distribution ----- Row 1 to 10 of 10
Command ==>                                           Scroll ==> PAGE
```

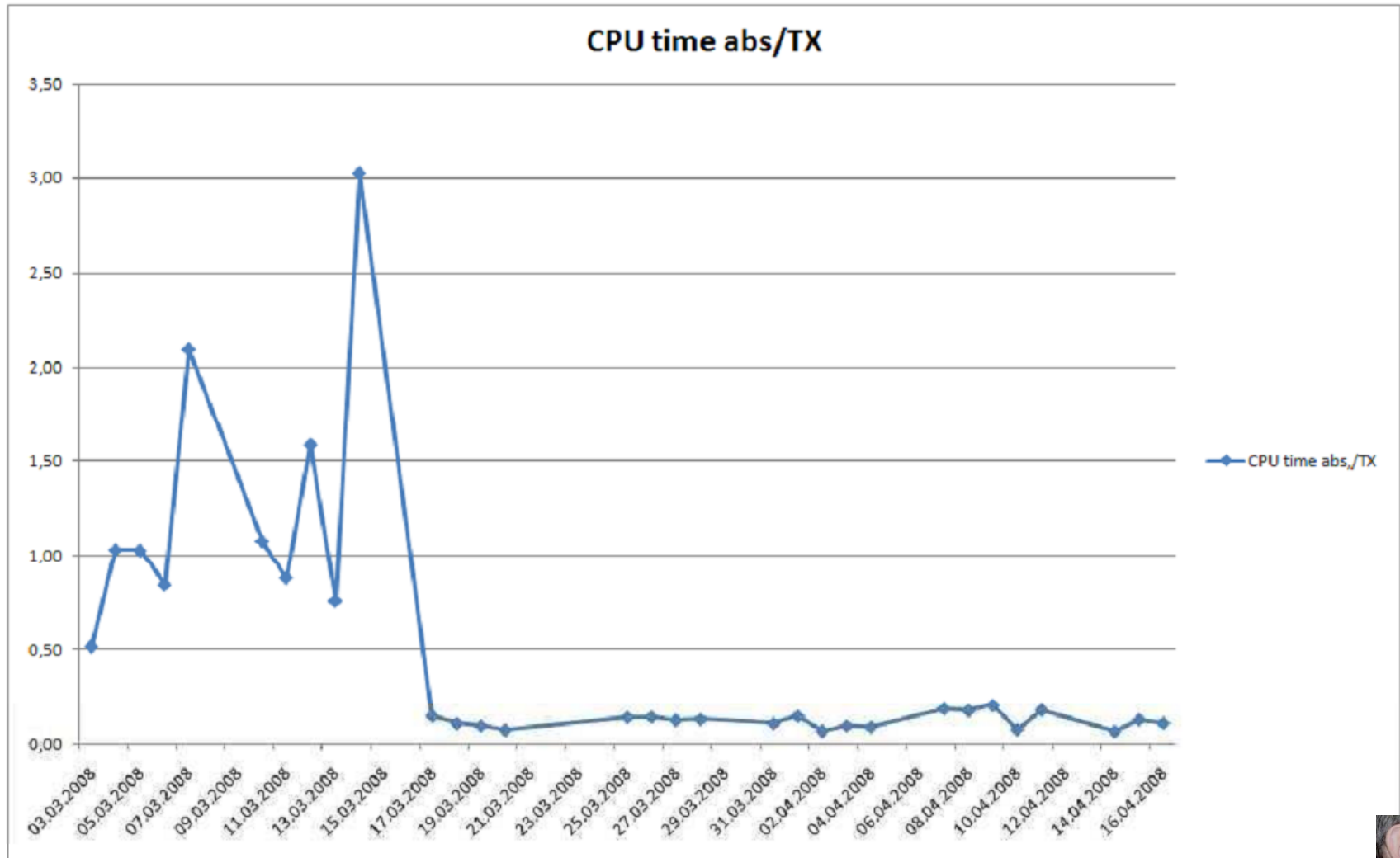
Line commands:

T - Table C - Column UR - Update runstats PST - Partition stats

I - Interpret RH - Runstats History

Table Name	Column Name	Percent	Value
*	*	*	*
-----	-----	-----	-----
UTEXAZU	FK_EX_ATTRIBUT_ID	31.48	0
UTEXAZU	FK_EX_ATTRIBUT_ID	4.24	1007
UTEXAZU	FK_EX_ATTRIBUT_ID	4.24	1002
UTEXAZU	FK_EX_ATTRIBUT_ID	4.24	1001
UTEXAZU	FK_EX_ATTRIBUT_ID	4.23	1029
UTEXAZU	FK_EX_ATTRIBUT_ID	4.23	1009
UTEXAZU	FK_EX_ATTRIBUT_ID	4.24	1004
UTEXAZU	FK_EX_ATTRIBUT_ID	4.24	1003
UTEXAZU	FK_EX_ATTRIBUT_ID	4.24	1027
UTEXAZU	FK_EX_ATTRIBUT_ID	4.24	1026
***** END OF DB2 DATA *****			

## DB2-Indexdesign – Beispiel Versicherung 2008 – 5



## Isolation Level

---

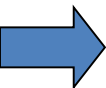
- CS – Cursor Stability
  - höchste Datenintegrität ohne „optimistic currency control“ bei „dynamic scrollable cursors“
- UR – Uncommitted Read
  - auch „dirty read“ genannt
  - geht nicht bei  
DELETE, UPDATE, INSERT, MERGE  
CURSOR ... FOR UPDATE
  - Sollte *\_immer\_* als Möglichkeit in Betracht gezogen werden!!!



## Isolation Level

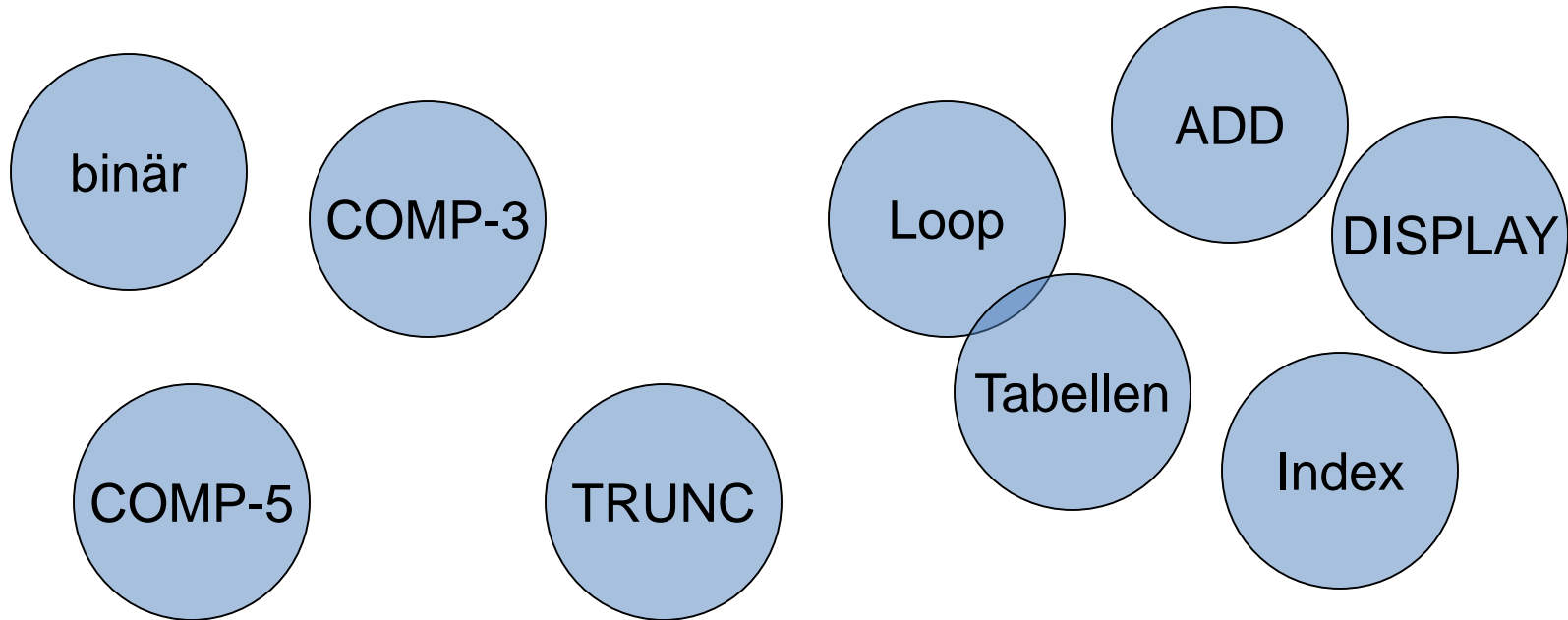
---

- RR – Repeatable Read
  - mehrfaches Lesen von Rows oder Pages
  - Jede benutzte Page wird gelockt, selbst wenn sie \*nicht\* den Predicates genügt.
  - \*kein\* paralleler Update erlaubt
- RS – Read Stability
  - mehrfaches Lesen von Rows oder Pages
  - Jede benutzte Page wird gelockt, selbst wenn sie \*nicht\* den Predicates genügt.
  - \*paralleler Update teilweise erlaubt
  - Gelockt werden Rows bzw. Pages, die Stage 1 und Stage 2 erfüllen (und keine anderen).

- 
- Vorstellung und Einführung
  - Optimierungen – Beispiele und Potential
  - Strobe – Handling
  - Vorgehensweise für die Analyse (Strobe)
  - Vorgehensweise für die Analyse (iStrobe)
  - Modellierung und DB2-Zugriffe
  -  • COBOL–Felder – COBOL-Befehle
  - Auswirkungen von Optionen – COBOL – LE
  - Praxisbeispiele – Diskussion – Austausch

## Begriffe

---



## Felddefinitionen

---

- Binärfelder – BINARY
  - Halbwort S9(04) oder Vollwort S9(08) mit Vorzeichen
- gepackte Felder – PACKED-DECIMAL
  - auf Bytegrenzen achten (S9(n) mit n ungerade  $\leq 15$ )
- “normale Felder” – USAGE DISPLAY
  - **nicht** für Rechenoperationen verwenden (no Perform!!!)
- Tabellen
  - nur mit Indizes (INDEXED BY)
  - n-dim: Loopverarbeitung von rechts nach links

Bitte benutzen, denn  
Wartungsfreundlichkeit  
ist wichtiger als  
Performance!

- INITIALIZE

- jedes einzelne Feld wird auf Anfangswert gesetzt
- jedes einzelne Feld wird auf Anfangswert gesetzt
- innerhalb Schleifen möglichst unterlassen
- Hilfsfelder nutzen
- jedes schwierige Beispiel muss separat beurteilt werden, daher kein “Kochrezept” möglich

außer FILLER

- STRING/UNSTRING/INSPECT/SEARCH

- zieht hohen CPU-Verbrauch nach sich
- ab V4R1 wird es schneller – NO ☹ ☹ ☹ noch nicht
- ab V5R1 ist es schneller 😊

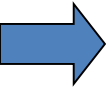
- **PERFORM VARYING**
  - Schleifenzähler binär definieren / gepackt
  - Begrenzer binär definieren / gepackt
  - bei Tabellenverarbeitung nur mit INDEX arbeiten
  - jederzeit auf Formatgleichheit achten
- **EVALUATE**
  - (leider wieder) häufigsten Fall zu Beginn codieren
- **Stufe 88**
  - sehr schnelle Verarbeitung nur bei CHAR!!

- Rechenoperationen
  - beteiligte Felder mit gleichen Längen
  - beteiligte Felder mit gleichem Format
- Vergleichsoperationen
  - beteiligte Felder mit gleichen Längen
  - beteiligte Felder mit gleichem Format
- Substr-Move
  - besser: `MOVE FELD-A(2:5) TO FELD-B (-> MVC)`
  - nicht: `MOVE FELD-A(2:N) TO FELD-B (-> MVCL)`
  - erste Zahl darf Variable sein

- MOVE
  - MVC ist schnell / MVCL ist langsam
  - MVC kann nur bis 256 Byte übertragen
  - bis Länge 768 werden MVCs generiert (aber nicht bei move spaces to zielfeld!!)
  - MVC / MVCL abhängig von **Zielfeld**

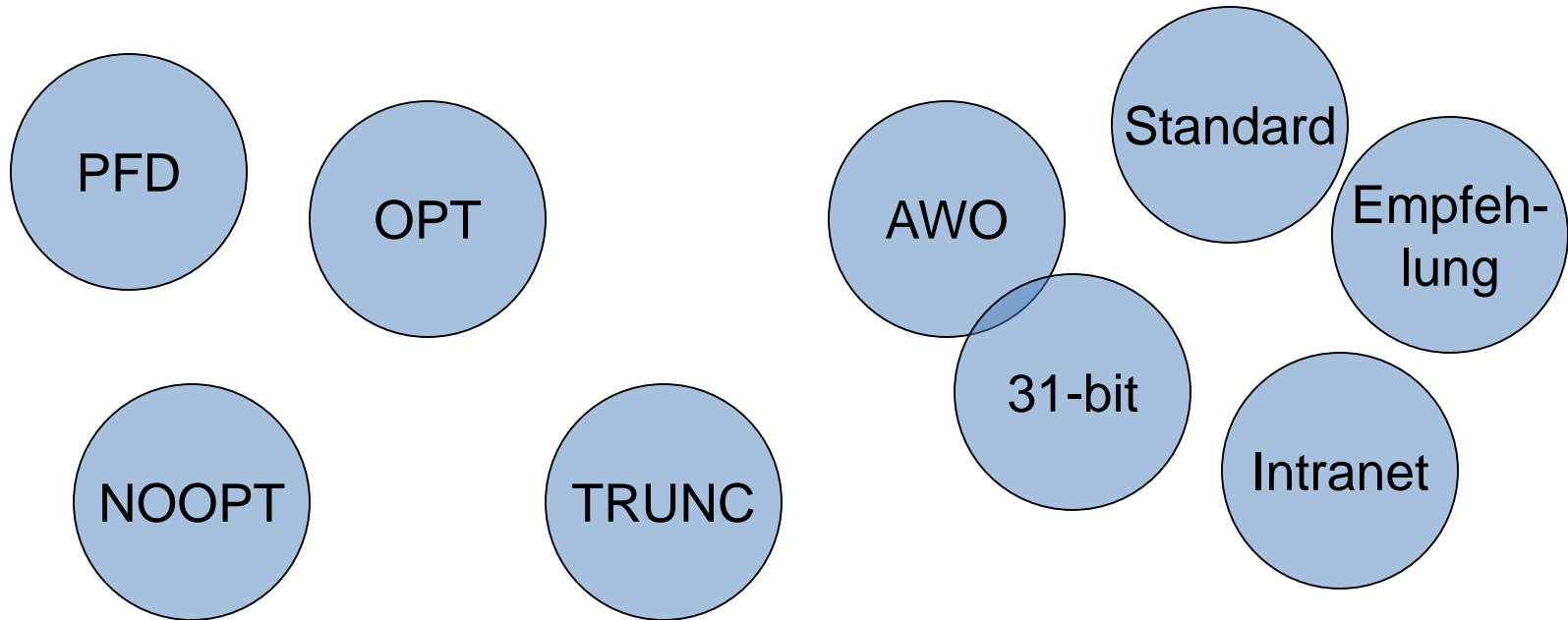




- 
- Vorstellung und Einführung
  - Optimierungen – Beispiele und Potential
  - Strobe – Handling
  - Vorgehensweise für die Analyse (Strobe)
  - Vorgehensweise für die Analyse (iStrobe)
  - Modellierung und DB2-Zugriffe
  - COBOL–Felder – COBOL-Befehle
  -  • Auswirkungen von Optionen – COBOL – LE
  - Praxisbeispiele – Diskussion – Austausch

## Begriffe

---



# COBOL Compile Options

„gute“ und „schlechte“ Options in Auswahl

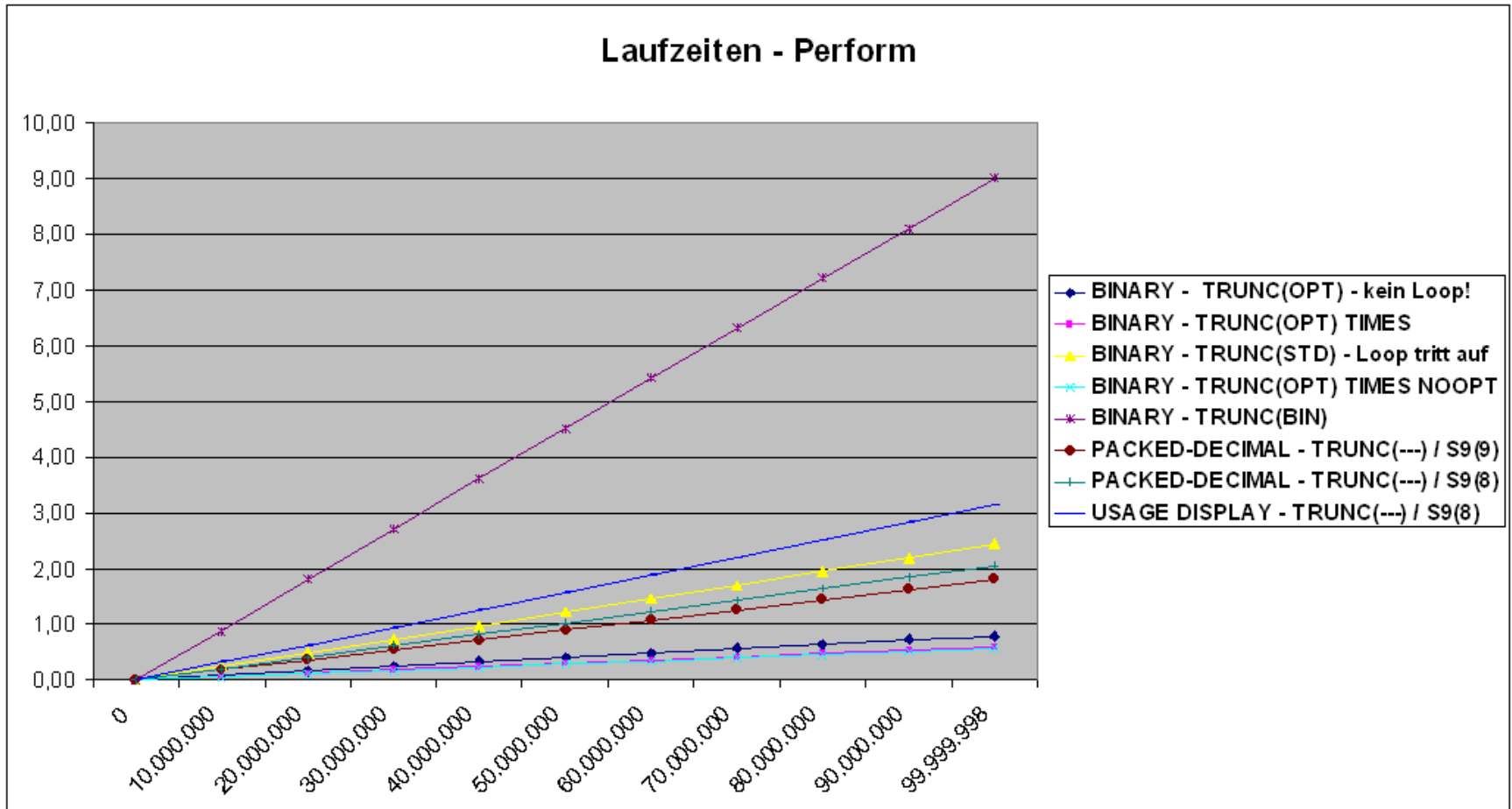
Notation: Standard Empfehlung

- → **NOOPT** | OPT(STD) | OPT(FULL)
- → TRUNC(OPT) | TRUNC(BIN) | TRUNC(STD)
- → NUMPROC(PFD) | NUMPROC(NOPFD)
- AMODE(24) | AMODE(31)
- DATA(24) | DATA(31)
- RMODE(24) | RMODE(ANY)
- NOSSRANGE | SSRANGE
- NOTEST | TEST

PFD = preferred sign

SSRANGE ist super für Test

- Ergebnisse

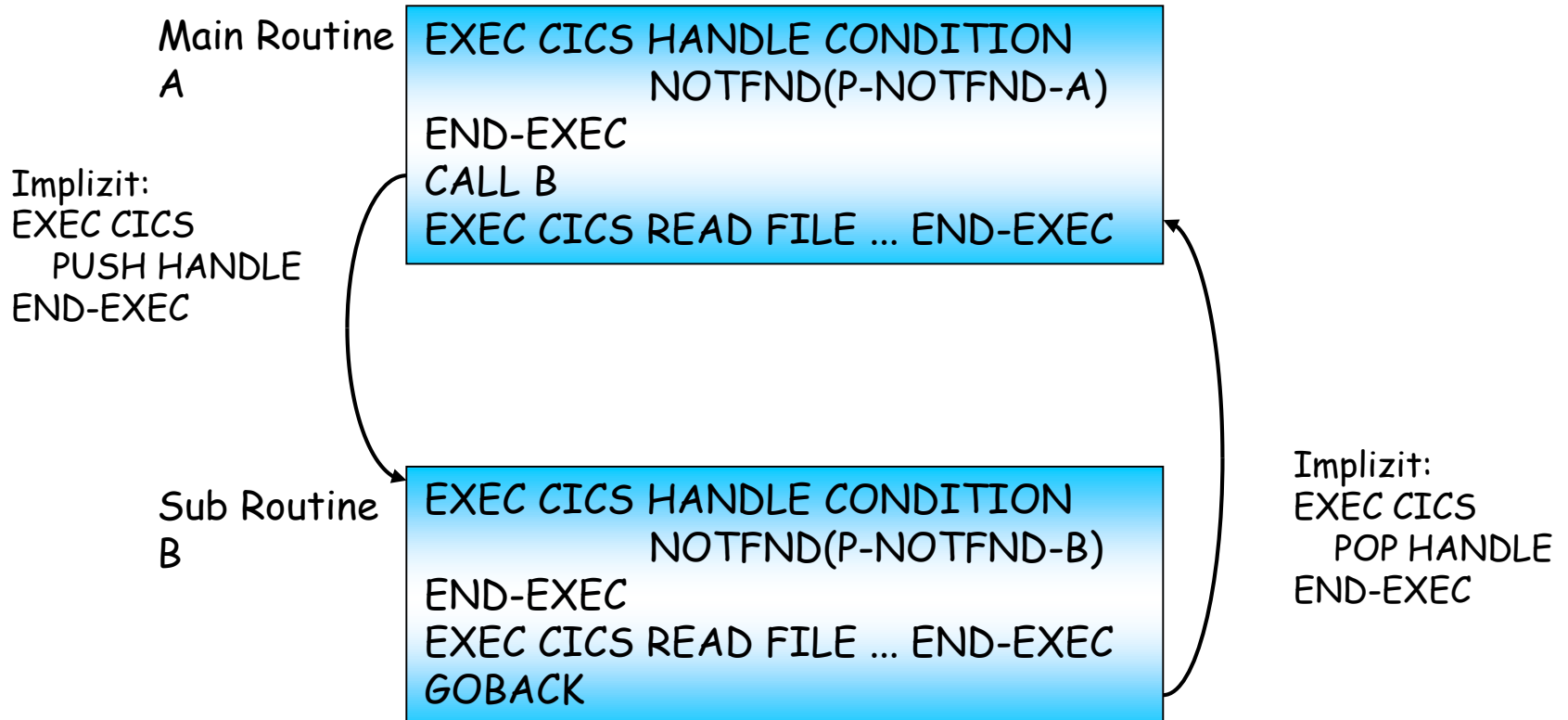


- Der Weg in Richtung 64-bit-Adressierung muss u.a. wegen der wachsenden Datenmengen konsequent verfolgt werden!
- LE-Option ALL31(ON) spart laut IBM ca. 3% der gesamten CPU-Last. Diese Option kann nicht gesetzt werden, so lange noch Anwendungsprogramme below-the-line laufen müssen.

# LE Options

## CBLPUSHPOP(ON) – 1

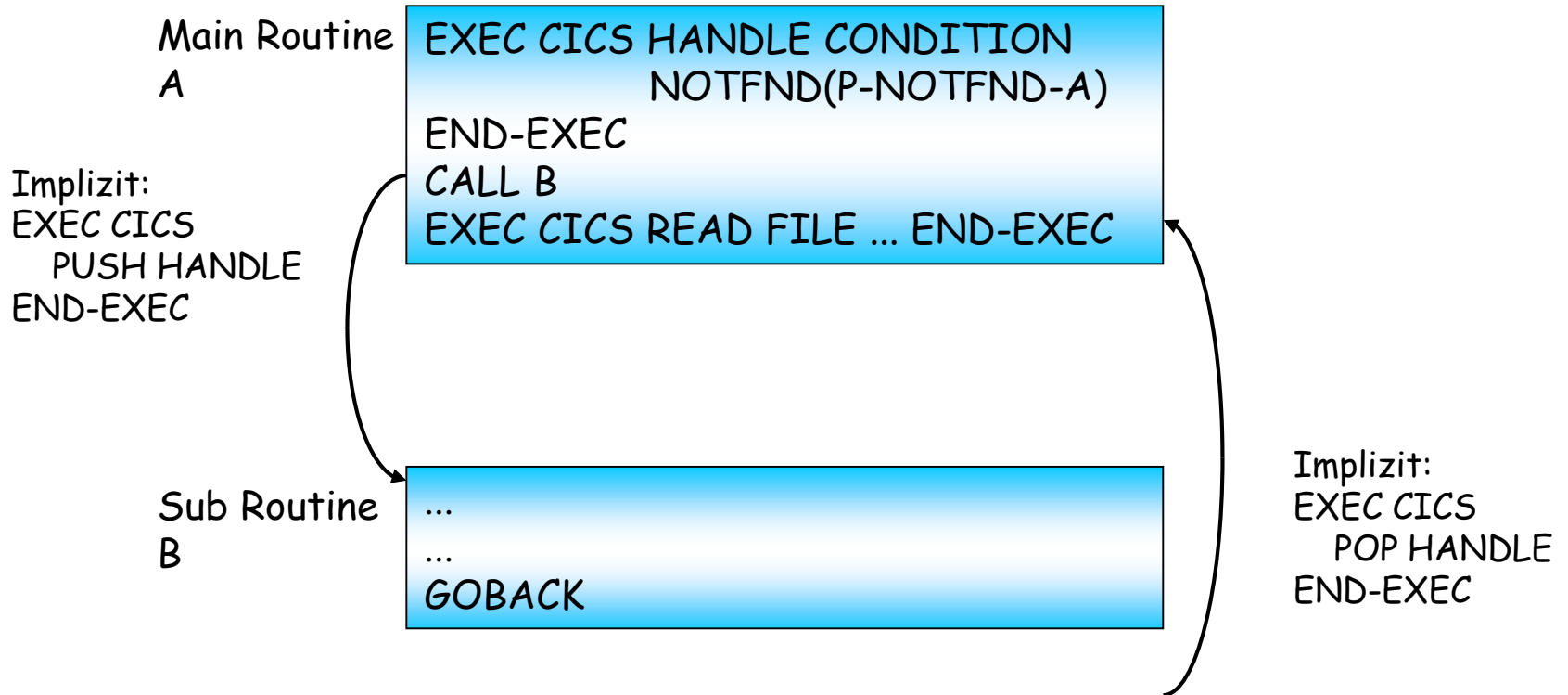
---



# LE Options

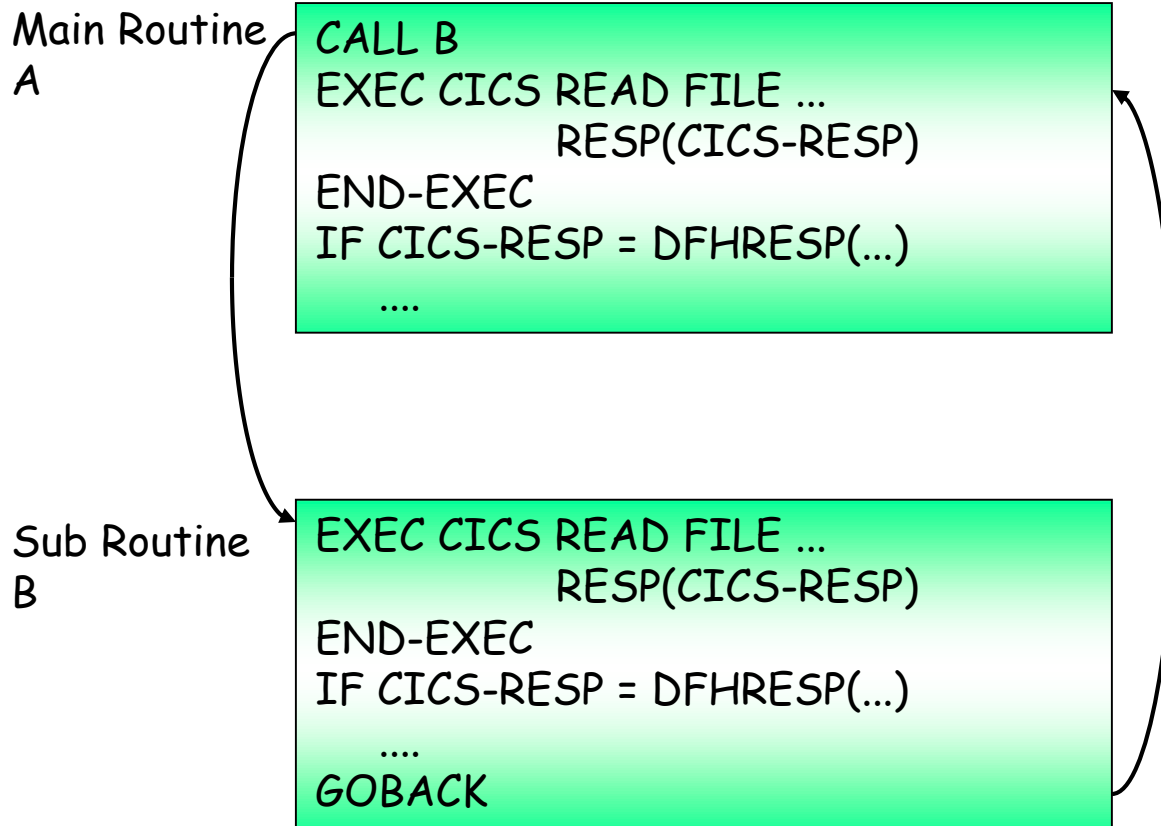
## CBLPUSHPOP(ON) – 2

---



# LE Options

## CBLPUSHPOP(OFF)



Voraussetzung:

- kein Handle Condition
- kein Handle Abend
- kein Handle AID

dringende Empfehlung:

- CICS-Commands mit Resp-Option
- LE-Condition-Handling

CBLPUSHPOP: Danke an Rita Backstein





- Vorstellung und Einführung
- Optimierungen – Beispiele und Potential
- Strobe – Handling
- Vorgehensweise für die Analyse (Strobe)
- Vorgehensweise für die Analyse (iStrobe)
- Modellierung und DB2-Zugriffe
- COBOL–Felder – COBOL-Befehle
- Auswirkungen von Optionen – COBOL – LE
- ➔ • Praxisbeispiele – Diskussion – Austausch

