



Oracle InMemory Application for reduced latency in maintenance processes

Oracle Open World, San Francisco, 2017

Agenda

1 Who are we

2 Who is my employer

3 What's the customers business

4 Fundamentals

5 Problem

6 Solution

Who are we



- Thorsten Pensky
- Age 48
- First contact with Oracle in 1993 (Oracle V6, SQL*Forms 3)
 - Development of warehouse management systems and the connection to SPCs for control of flow of materials
 - Guidance systems for pallet transporter
 - Voice controlled picking of goods
- First DBA course for Oracle 9i
- Actually responsible for databases at Lufthansa
 - Provisioning/Customizing/Maintenance/Optimizing
 - Invoice
 - Human Resource Planing
- Hobbies: Music, travelling & scuba diving

Who are we

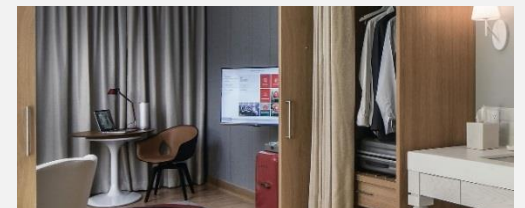


- Sonja Meyer
- Age formerly known ;-)
- First contact with Oracle in 2009 (Oracle 10g, PL/SQL)
 - 2nd level support for Siebel and self-developed applications on ORACLE database
 - Development of warehouse management system and forecast
 - Business intelligence
- Started at ORACLE in 2009 in consulting expert service responsible for MAA and performance tuning
- Actually working for PreSales in Germany as an IT cloud architect Technology responsible for large accounts in cloud architecture, and project lead POCs (EXADATA and Database IN-Memory) and for crazy ideas
- Hobbies: Music, travelling, reading, beachvolleyball & running

Who is my employer



- Lufthansa Industry Solutions
 - Subsidiary of Lufthansa Group
- IT-provider
- > 1300 employees
- 10 sites
- > 200 customers
 - MRO
 - Transport & logistics
 - Manufacturing
 - Automotive
 - Energy
 - Media
 - Healthcare



Who is my employer

- My business segment AB/M-E
- 19 colleagues
- Maintenance of
 - Oracle/MS SQL databases
 - Webmethods/servicebus
 - Openshift
- Administration/tuning/consulting
- Actually
 - 290 Oracle databases
 - 16 MS SQL databases
- New customer asking for service of 160 more Oracle databases



What's the customers business

- Primarily
 - Maintenance of 2000 aircrafts at more than 60 airports worldwide
 - Overhaul/repair of all components of 30 types of aircrafts
 - From coffee maker
 - To complete engines (40 types)
 - And landing gears
 - Operated 24/7/365
- Secondary
 - Support for private VIP, business and government aircrafts
 - Cabin upgrades
 - Maintenance
 - Painting

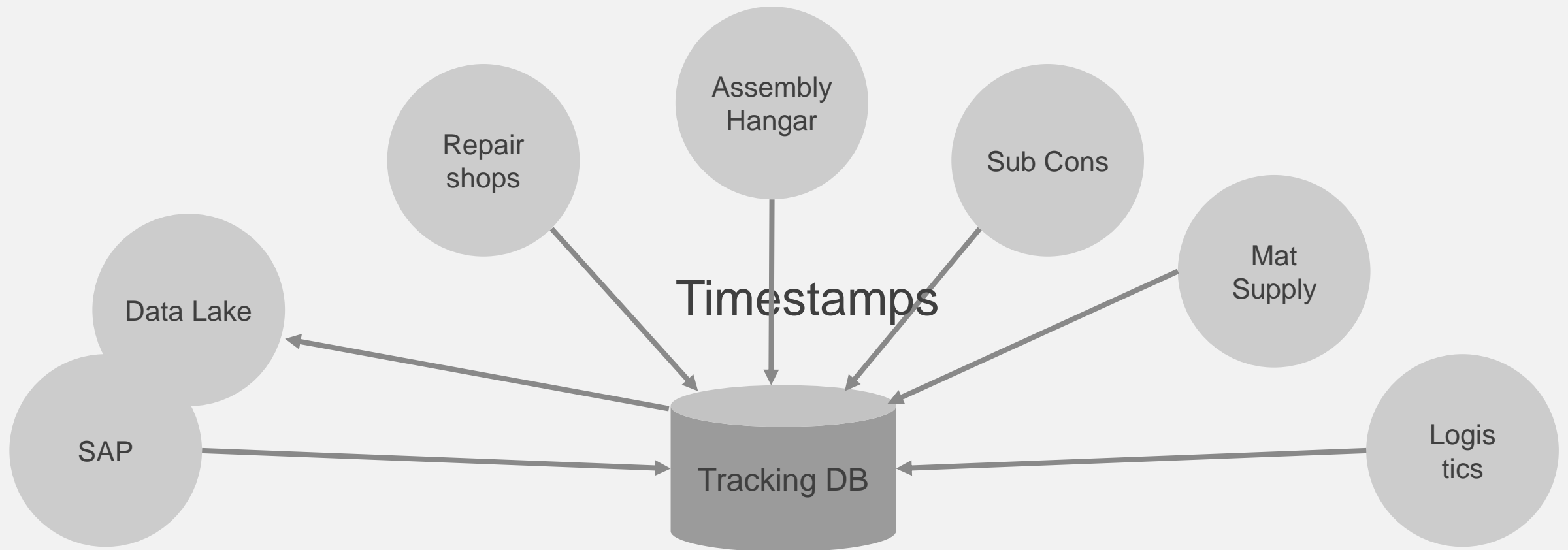
What's the customers business – the special segment

- Every 6–10 years a D-check is performed
- Takes 4–6 weeks and 30,000–50,000 man-hours of labor
- Airplane is nearly completely disassembled
- Every part is examined if it needs repair
- Repair is done in own workshops or by subcontractors

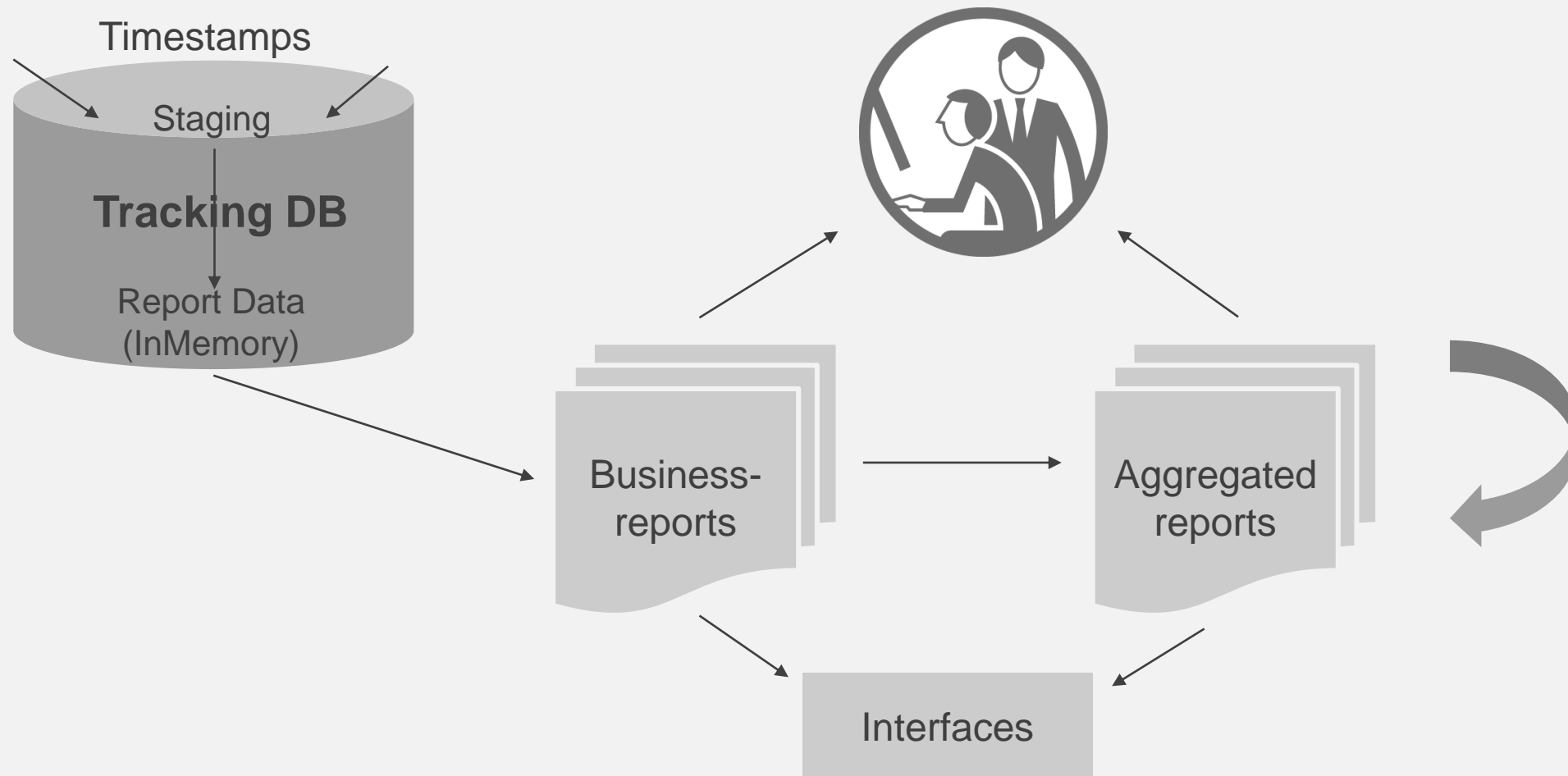
- Tracking is very important
 - Which component is send for repair
 - Where is it actually (transport, repair, warehouse)
 - Will it be usable again and when
 - If not, which component can be used instead



What's the customers business – the special segment

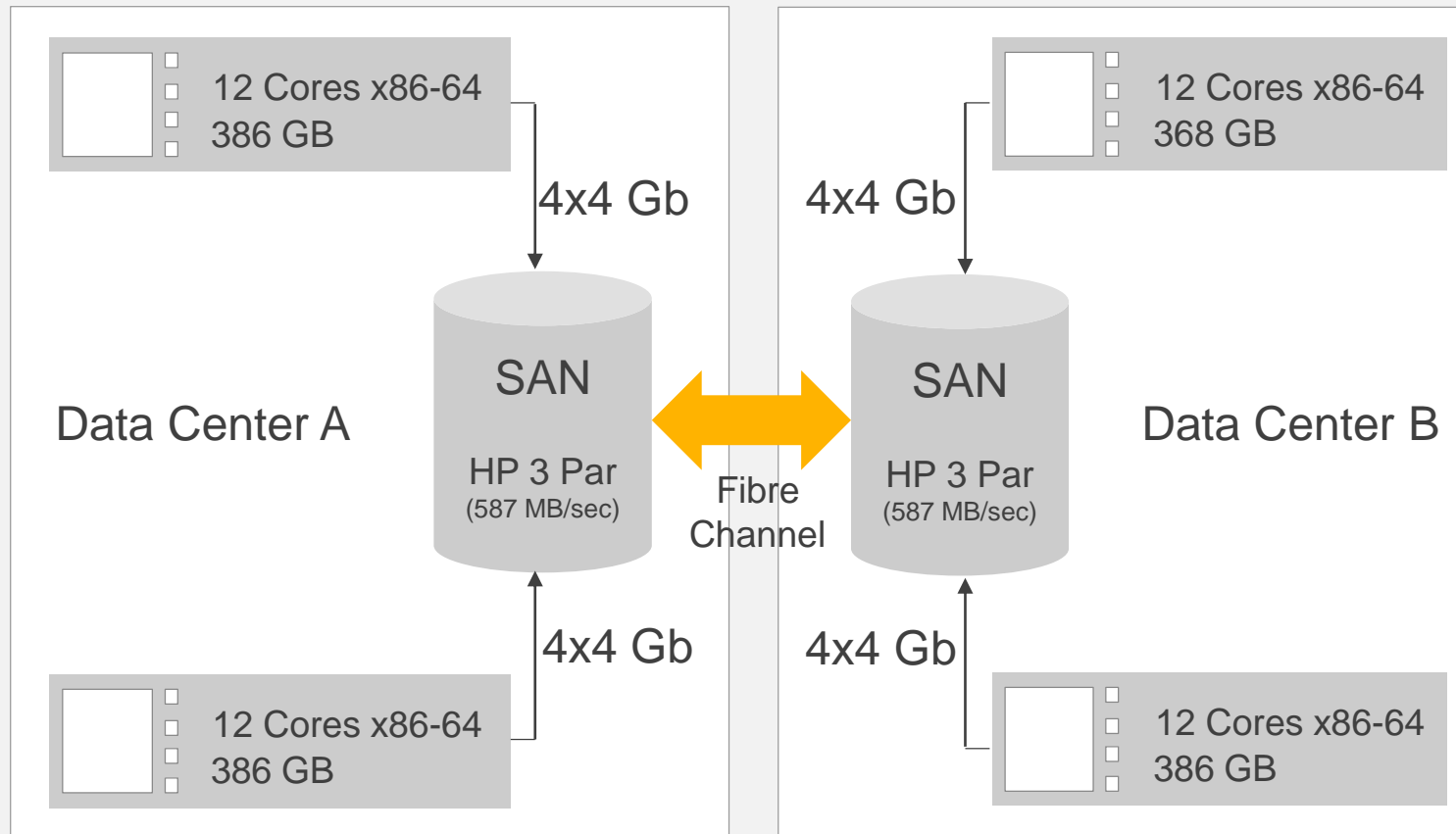


What's the customers business – the special segment



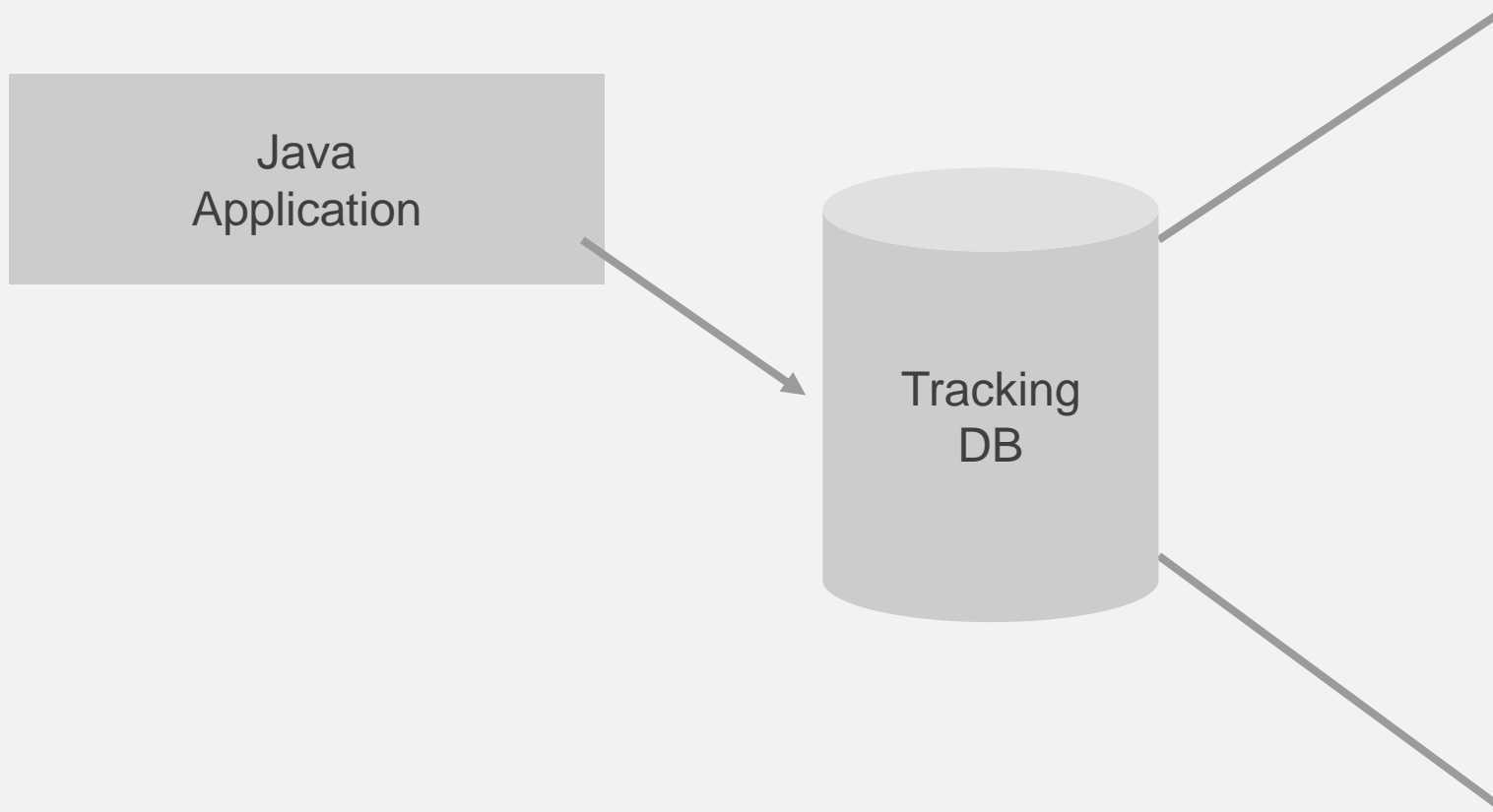
Fundamentals

RAC OneNode Cluster



- RedHat Enterprise 6 x86-64
- 110 Oracle Databases (11.2 & 12.1)
 - 38 Production
 - 72 Test
- Oracle Grid Infrastructure
- ASM / ACFS
- RAC One-Node
- Partitioning Option
- **In-Memory**








Fundamentals



- Oracle 11.2.0.4
- SGA 24 GB
- PGA 5 GB
- AMM

- Total Size 1.2 TB

Fundamentals

Customer Status Report											
 		Total Records: 8				  1 of 1  		Records per Page 10 			
Customer	Supplier	PO No	PO Item	PO P/N	PO S/N	PO Qty	Status	PO Date	Rcvd Qty	Rcvd P/N	Rcvd S/N
Customer A	Supplier C	203599	1	1703M30	DF5739	1	In progress	15 Aug 17	1	1703M30	DF5739
Customer A	Supplier C	203668	1	500730-1		1	shipping	12 Aug 17	1	500730-1	
Customer C	Supplier E	203212	1	31DA10		1	ready to ship	10 Aug 17	1	31DA10	
Customer F	Supplier D	203781	1	9092M4	LJA443	2	In progress	18 Aug 17	2	9092M4	LJA443
Customer F	Supplier D	203781	2	1475M3	MDDA03	1	In progress	18 Aug 17	1	1475M3	MD23HC
Customer H	Supplier R	202355	1	6A76L4		1	closed	02 Aug 17	1	6A76L4	DF5739
Customer K	Supplier A	204002	1	508P27	AL2344	1	Repair	23 Aug 17	1	508P27	AL2344
Customer M	Supplier G	204137	1	45-008	LY222	1	Open	30 Aug 17	1	45-008	LY222

Fundamentals

- Underlying view selects
 - 31 tables
 - 5 views
 - 3 materialized views
- > 2000 lines of code

```
SELECT
CASE
WHEN MD_SUBCON_INDICATOR = 'I'
AND SHOP_TST_TAKEOVER - MD_PROCESS_ID_START >= 0
THEN
ROUND (SHOP_TST_TAKEOVER - MD_PROCESS_ID_START, 1)
WHEN MD_SUBCON_INDICATOR IN ('F', 'N')
AND SC_TAKEOVER_SUBCON - MD_PROCESS_ID_START >= 0
THEN
ROUND (SC_TAKEOVER_SUBCON - MD_PROCESS_ID_START, 1)
ELSE
NULL
END
BB_TAT_INBOUND

CASE
WHEN SHOP_TST_TAKEOVER - MD_PROCESS_ID_START >= 0
THEN
ROUND (SHOP_TST_TAKEOVER - MD_PROCESS_ID_START, 1)
ELSE
NULL
END
BB_TAT_LOG_CUST_TO_LHT_SHOP

.....

AND outb_hub.process_id = vcp.process_id)
LEFT JOIN br_ext_trans outb_hub_et
ON (outb_hub_et.ext_trans_no = outb_hub.ext_trans_seg1)
/* == end == */

/* == Subcon == */
LEFT JOIN v_contracted_webi ---V_AR_SUBCON_BY_SVC_ORDER_NO
---MV_AR_SUBCON_BY_SVC_ORDER_NO
---group by service_order
sco
ON ( sco.SERVICE_ORDER_NO = vcp.svc_order_no
AND NVL (sco.PURCHASE_ORDER_DELETION, 'N/A') != 'L')
/* == end == */

WHERE 1 = 1
/* Generelle Kriterien */
---and vcp.business_process=CLOSED_LOOP'
AND vcp.business_process IN ('CLOSED_LOOP', 'OPEN_LOOP')
AND vcp.TST_PROCESS_CREATED > SYSDATE - 3 * 365
---and vcp.cust_no=7922' and vcp.tst_shipping_outbound > sysdate-90
---and o.SVC_ORDER_ACTIVITY_TYPE=MOD'
---and VCP.PROCESS_ID_ACTIVE in (select md_lht_reference from mv_ar_csr_complete_webi))
---and vcp.process_id_active in ('90973477','61200487','61198381','61197041','61197747','90876772','90874905','90874922','90800004')
---and vcp.mat_reference='1675363'
---and vcp.process_id_active in ('61200487')
---and vcp.zid='392237682'
---and vcp.mro_id='100000686171'
---and vcp.mro_id='100000127168' and vcp.process_id_active='61125790'
---and vcp.outb_shipment_order_no='5175800010'
) bd;
```

Problem

Runtime of views used in daily routine

View-Content	Runtime 11g (Minutes)*	Runtime 12c (Minutes)**
How long takes repair of the engine	3:47	4:46
How long takes the transport of the engine	2:45	0:27
Timereport of component maintenance	0:06	0:10
Complete report of customer components	4:46	3:00
Customer sends back spare component	8:27	6:37
Average processing time over last 3 years	3:50	- ***
Performancereport	10:00	- ***

* 11.2.0.4 ** 12.1.0.2.2 *** lost in space

Tests done on a single node without other databases to avoid any interference

Solution

- Next step was a proof of concept: 31st March 2015–2nd April 2015
 - apply actual patch set #20329440
(DATABASE PATCH FOR ENGINEERED SYSTEMS AND DB IN-MEMORY 12.1.0.2.5)
 - establish huge pages on server
 - enable inmemory in database
 - SGA_TARGET = 150G (formerly 24G)
 - INMEMORY_SIZE = 120G
 - MEMORY_TARGET = 0 (no AMM due to use of HugePages)
 - INMEMORY_MAX_POPULATE_SERVERS = 12 (#cores)
 - lookup tables used in the reports (no IM advisor because we knew what we did)
 - configured tables for inmemory
- Check changes in behaviour

Starting from April 2016
the prior Database Bundles
will now be called
“Proactive Bundle Patches”
MOS 2285558.1

Solution

Examples of tables inmemory (total 76 tables)

MEMCOMPRESS FOR QUERY LOW PRIORITY HIGH

SEGMENT_NAME	TYPE	IM_SIZE	BYTES	COMP (%)	FACTOR
BR_LTL_REQUEST_TIMETABLE_FINAL	TABLE	830.930.944	1.940.291.584	57,17	2,34
AR_CUSTOMER_PROCESS	TABLE	960.057.750	3.685.621.760	73,95	3,84
BR_CLARIF_RECV	TABLE	339.345.408	786.808.832	56,87	2,32
BR_BUSINESS_PROC_CLARIF_GROUP	TABLE	1.179.648	1.695.744	30,43	1,44
BR_ORDER_STATUS	TABLE	2.983.526.400	5.629.870.080	47,01	1,89
AR_ORD_PPS_MM	TABLE	181.141.504	226.967.552	20,19	1,25
AR_TCS_IN	TABLE	316.604.416	402.440.192	21,33	1,27
BR_LOCATION_BOOK	TABLE	1.128.595.456	1.540.235.264	26,73	1,36

Factor varies from 1.08 up to 3.84

Solution

Attention

SEGMENT_NAME	TYPE	IM_SIZE	BYTES	COMP (%)	FACTOR
BR_ORDER_ATTACH	TABLE	1.179.648	106.496	-1107,69	-11,08
BR_ORDER_CLARIF_GROUP	TABLE	1.179.648	671.744	-175,61	-1,76

Explanation:

InMemory has a little bit of overhead for every table

-> **Small tables don't need to be inmemory**

Solution

But it can also happen to larger tables

SEGMENT_NAME	TYPE	IM_SIZE	BYTES	COMP (%)	FACTOR
AR_BUSINESS_PROC_CLARIF	TABLE	182.190.080	176.832.512	103,03	-1,03
BR_MATERIAL_RETURNS	TABLE	72.548.352	71.737.344	101,13	-1,01

Explanation:

No real explanation. After installation of two more bundle patches the issue vanished.

-> Keep your database up-to-date

Solution

Runtime of views used in daily routine in minutes (12c only)

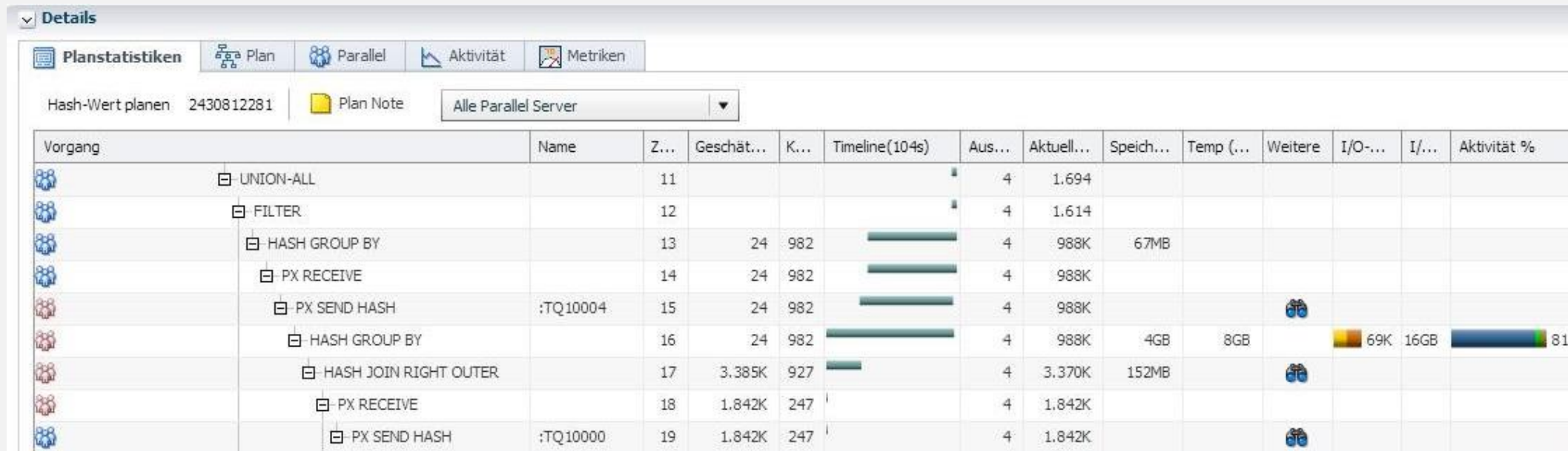
View-Content	w/o IM	Serial with Indexes	Serial w/o Indexes	PQ4 with Indexes	PQ4 w/o Indexes
How long takes repair of an engine	4:46	00:54	00:57	00:04	00:38
How long takes the transport of an engine	0:27	00:18	00:18	08:17	09:12
Timereport of component maintenance	0:10	00:05	00:27	00:15	00:13
Complete report of customer components	3:00	04:30	- **	00:54	- **
Customer sends back spare component	6:37	00:19	00:06	00:20	00:03
Average processing time over last 3 years	- *	00:44	00:42	00:12	00:12
Performancereport	- *	00:06	00:21	00:15	00:03

* lost in space ** cancelled due to runtime - see next slide

Tests done on a single node without other databases to avoid any interference

Solution

Runtime problem of view



Vorgang	Name	Z...	Geschät...	K...	Timeline(104s)	Aus...	Aktuell...	Speich...	Temp (...)	Weitere	I/O-...	I/...	Aktivität %
UNION-ALL		11				4	1.694						
FILTER		12				4	1.614						
HASH GROUP BY		13	24	982		4	988K	67MB					
PX RECEIVE		14	24	982		4	988K						
PX SEND HASH	:TQ10004	15	24	982		4	988K						
HASH GROUP BY		16	24	982		4	988K	4GB	8GB		69K	16GB	81
HASH JOIN RIGHT OUTER		17	3.385K	927		4	3.370K	152MB					
PX RECEIVE		18	1.842K	247		4	1.842K						
PX SEND HASH	:TQ10000	19	1.842K	247		4	1.842K						

Exceptional use of temp tablespace with “hash group by”

Fixed with Bundle Patch 6

Solution

Runtime of views used in daily routine in minutes (12c only) (PARALLEL_DEGREE_POLICY=AUTO)

View-Content	w/o IM	Auto DOP with Index	Auto DOP w/o Index	Auto DOP with Index *
How long takes repair of the engine	4:46	00:12 (PQ8)	00:26 (PQ8)	00:02 (PQ8)
How long takes the transport of the engine	0:27	00:53 (S)	00:55 (S)	00:54 (S)
Timereport of component maintenance	0:10	00:08 (S)	00:27 (S)	00:04 (S)
Complete report of customer components	3:00	- **	- **	- **
Customer sends back spare component	6:37	00:30 (S)	00:06 (S)	00:12 (S)
Average processing time over last 3 years	-	00:42 (S)	00:41 (S)	00:46 (S)
Performancereport	-	00:25 (PQ8)	00:01 (PQ8)	00:19 (PQ8)

* new statistics for test reasons ** remember previous slide, it's the same trouble

Tests done on a single node without other databases to avoid any interference

Solution

Fastest runtimes of views in minutes

View-Content	11g	Fastest time	Factor	Final (no PQ)
How long takes repair of the engine	3:47	0:02 Auto DOP with Ind (PQ8)	113	4.2
How long takes the transport of the engine	2:45	0:18 Serial with or w/o Indexes	9.2	9.2
Timereport of component maintenance	0:06	0:04 Auto DOP with Ind (PQ8)	1.5	1.2
Complete report of customer components	4:46	0:54 With Indexes (PQ4)**	5.3	1.1
Customer sends back spare component	8:27	0:03 Without Indexes (PQ4)	169	26.7
Average processing time over last 3 years	3:50	0:12 Serial with or w/o Indexes	19	5.5
Performancereport	10:00	0:01 Auto DOP w/o Ind (PQ8)	600	100

** on stardate -306752.4 we lost contact

Tests done on a single node without other databases to avoid any interference

Solution

Surprise looking into v\$im_segments !!!

SEGMENT_NAME	TYPE	IM_SIZE	BYTES	BYTES NOT POPULATED	STATUS
BR_LTL_REQUEST_TIMETABLE_FINAL	TABLE	830.930.944	1.940.291.584	0	COMPLETED
AR_CUSTOMER_PROCESS	TABLE	108.920.832	3.685.621.760	2.418.122.752	COMPLETED
BR_LOCATION_BOOK	TABLE	1.128.595.456	1.540.235.264	0	COMPLETED
BR_SPLITTING	TABLE	207.028.224	485.998.592	225.509.376	COMPLETED

Explanation:

The status shown in the view is partially wrong.
This was found to be a bug, which was resolved in PBP 161018.

-> **Keep your database up-to-date**

Solution

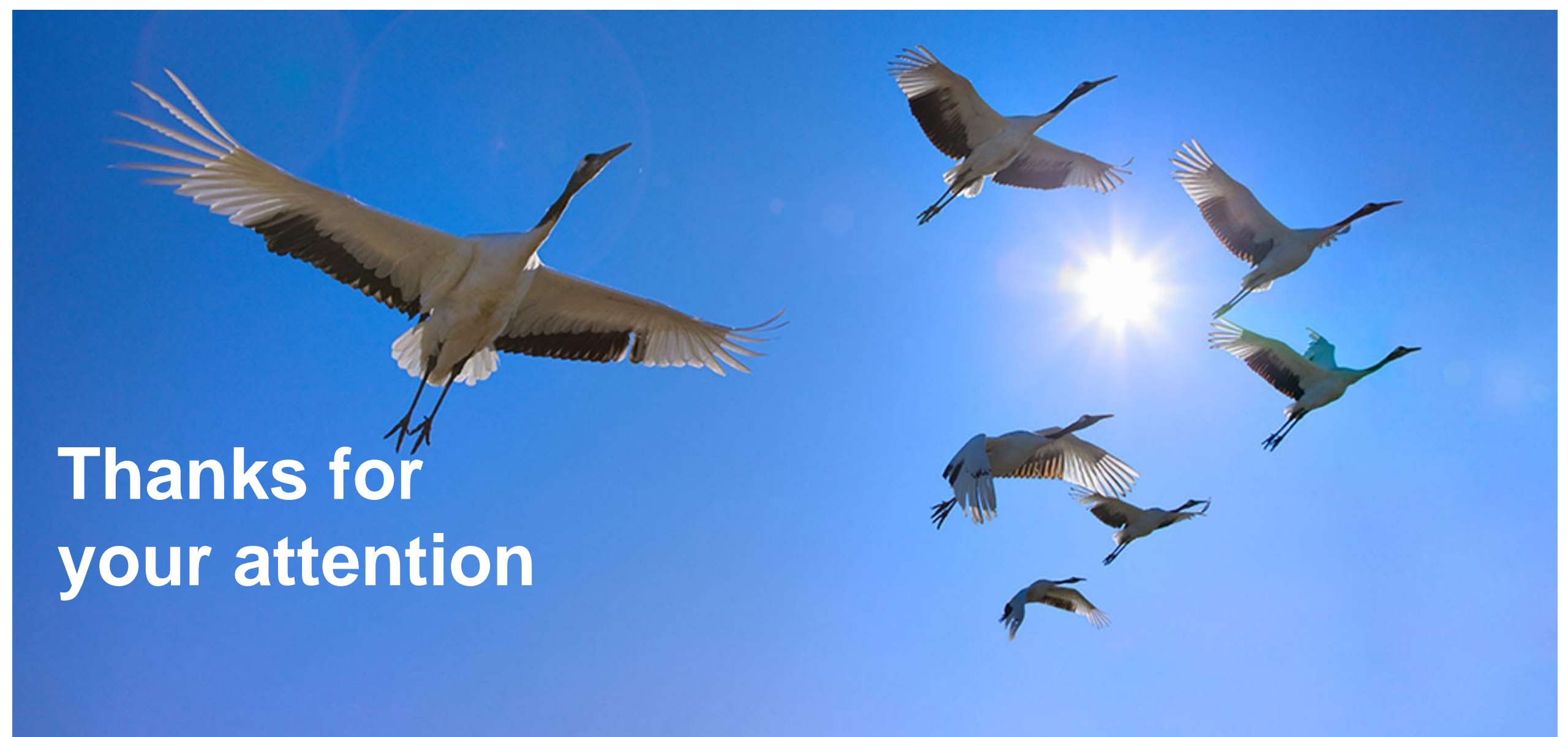
- Conclusion
 - **Fact: In-memory solved our problems**
 - Due to shared infrastructure we started with no parallelism
 - Runtimes could be reduced by an average of factor 8 to 10
 - I/O was reduced which is a benefit for our shared servers
 - Diskspace was freed by deleting indexes
 - Today even dynamic SQL is allowed to users (self-service BI tools)

Solution

- Where are we today
 - new storage system is used (IBM XIV) (avg. 900MB/sec)
 - storage is connected to servers via 16 GB adapters
 - RAM on every node increased to 768 GB
- starting with customers production system we put all report tables inmemory (76 tables)
 - beginning with an inmemory_target of 80 GB today we use 110 GB due to increased amount of data and new tables (act. 88)
- more compression (QUERY HIGH) due to increased amount of data -> although no speed impact
- two more databases starting to use inmemory; three more databases to come



Any questions left?



Thanks for
your attention