

ORACLE®



ORACLE®

Migration to Linux on IBM System z for Pros

Thomas Niewel
Principal Sales Consultant
Oracle Deutschland GmbH

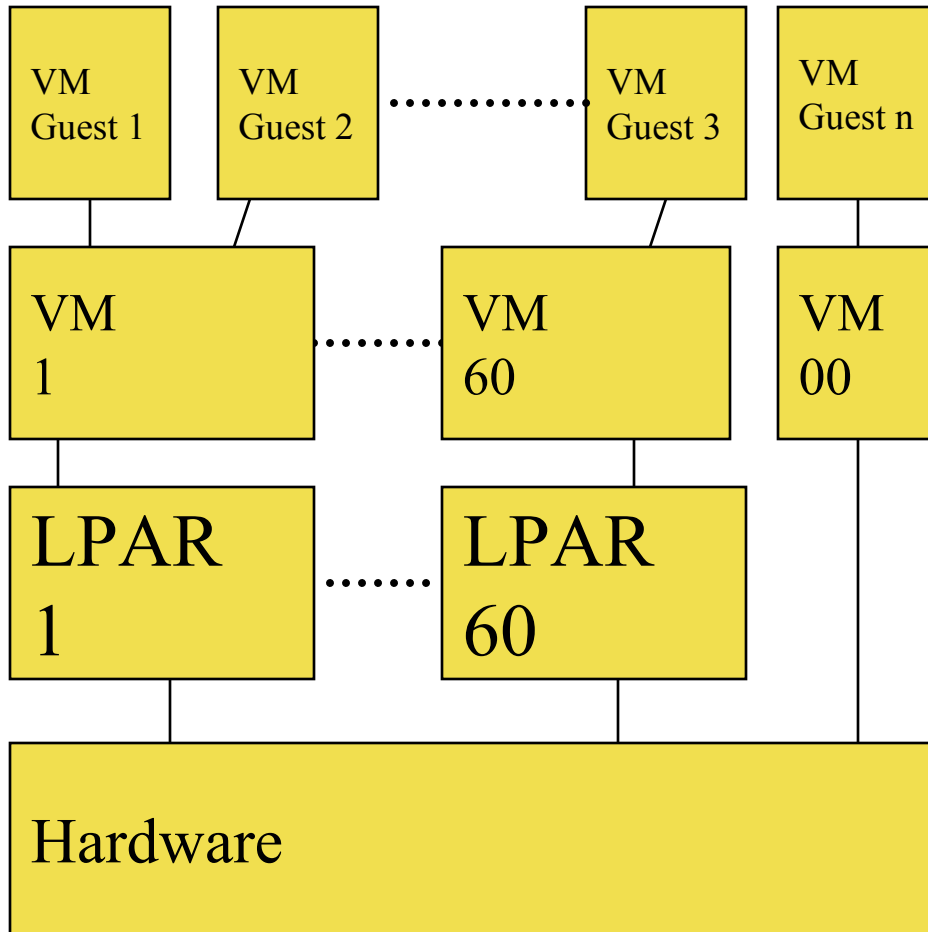
Agenda

- Linux on System z configuration variations
- Load profile determination of the source environment
- Characterset differences
- Schema and data migration
- Impact on applications
- Best practices

Agenda

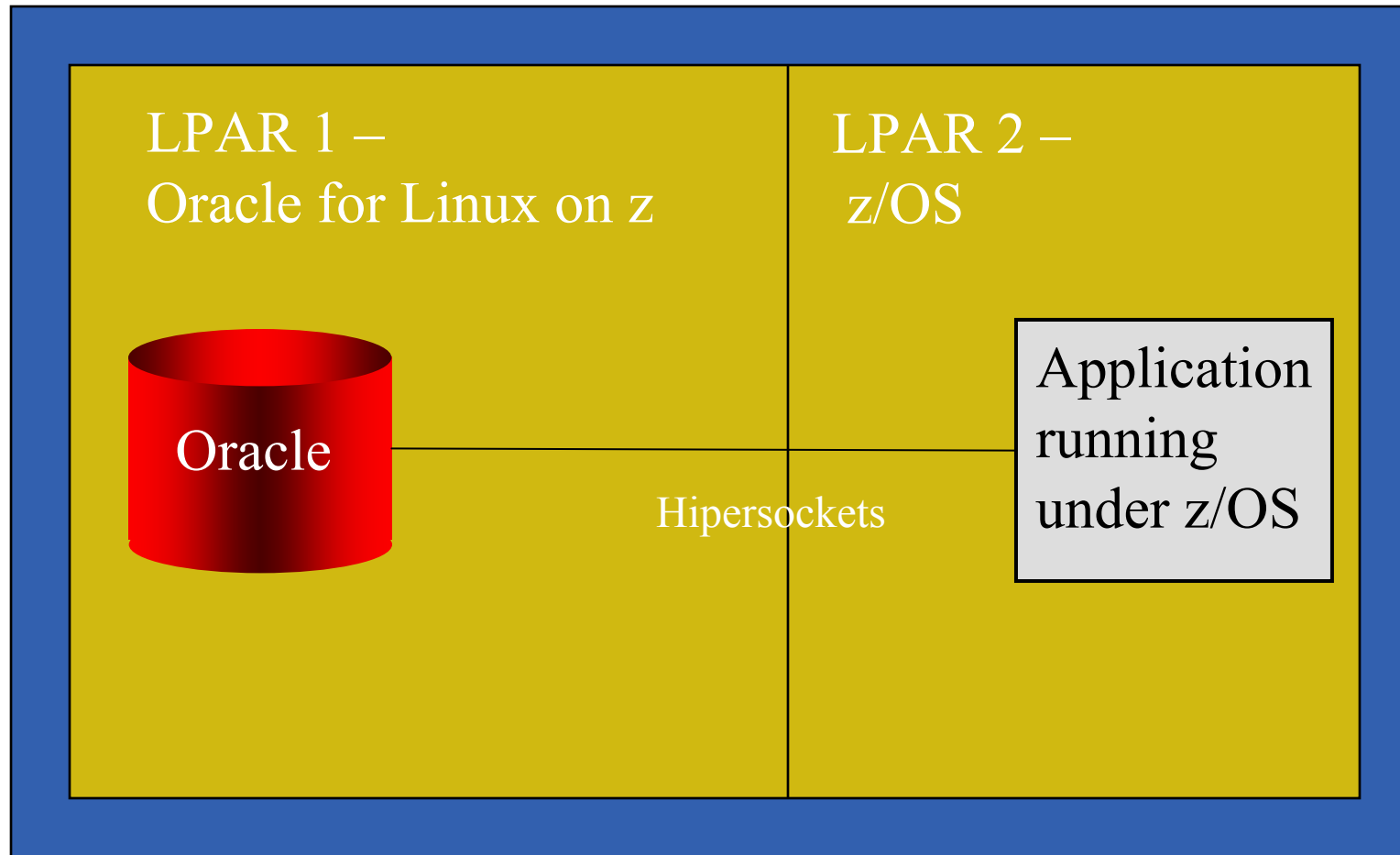
- Linux on System z configuration variations
- Load profile determination of the source environment
- Characterset differences
- Schema and data migration
- Impact on applications
- Best practices

System z configuration examples



- Virtual machines
 - VM-Guests
 - Linux for z
 - 1 .. n
- z/VM Operating System
- Logical partitions
 - 1 .. 60 LPAR's
- Basic Mode

Hipersockets – Example in a z/OS client environment



Conclusion

- Linux for z with an Oracle database can be configured in several variations
- An exact sizing of the target environment has to be performed
- The resource usage of the source environment has to be determined

Agenda

- Linux on System z configuration variations
- Load profile determination of the source environment
- Characterset differences
- Schema and data migration
- Impact on applications
- Best practices

Determine the resource usage of the source environment


- Oracle
 - Oracle Enterprise Manager Grid Control
 - AWR/statspack reports
- Unix
 - sar
 - iostat
- z/OS
 - RMF
 - SDSF

Example: I/O determined by AWR report

Load profile

~~~~~

|                  | Per Second    | Per Transaction |
|------------------|---------------|-----------------|
|                  | -----         | -----           |
| Redo size:       | 18,884,730.34 | 72,451.84       |
| Logical reads:   | 26,685.24     | 102.38          |
| Block changes:   | 16,810.84     | 64.50           |
| Physical reads:  | 272.99        | 1.05            |
| Physical writes: | 2,170.07      | 8.33            |
| User calls:      | 2,597.49      | 9.97            |
| Parses:          | 521.30        | 2.00            |
| Hard parses:     | 0.05          | 0.00            |
| Sorts:           | 1.37          | 0.01            |
| Logons:          | 0.05          | 0.00            |
| Executes:        | 523.53        | 2.01            |
| Transactions:    | 260.65        |                 |



# Example:

## Avg CPU usage determined by AWR report

| Statistic                        | Total   | per Second | per Trans |
|----------------------------------|---------|------------|-----------|
| -----                            | -----   | -----      | -----     |
| CPU used by this session         | 40,637  | 44.0       | 0.2       |
| CPU used when call started       | 40,431  | 43.7       | 0.2       |
| CR blocks created                | 6       | 0.0        | 0.0       |
| Cached Commit SCN referenced     | 476     | 0.5        | 0.0       |
| Commit SCN cached                | 1       | 0.0        | 0.0       |
| DB time                          | 572,703 | 619.5      | 2.4       |
| DBWR checkpoint buffers written  | 83,867  | 90.7       | 0.4       |
| DBWR checkpoints                 | 33      | 0.0        | 0.0       |
| DBWR fusion writes               | 2,253   | 2.4        | 0.0       |
| DBWR object drop buffers written | 0       | 0.0        | 0.0       |
| DBWR thread checkpoint buffers w | 19,530  | 21.1       | 0.1       |

# Agenda

- Linux on System z configuration variations
- Load profile determination of the source environment
- Characterset differences
- Schema and data migration
- Impact on applications
- Best practices

# Charactersets

- Unicode solves codepage mapping issues(see Metalink Note: 333489.1)
- Be careful if ASCII charactersets are used
- Example
  - Source database with an EBCDIC characterset WE8EBCDIC1047
  - Target database with an ASCII characterset WE8ISO8859P1

# Character set WE8EBCDIC1047

|    | 0         | 1         | 2              | 3         | 4              | 5        | 6              | 7              | 8              | 9              | A              | B              | C              | D              | E              | F              |
|----|-----------|-----------|----------------|-----------|----------------|----------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 0  | CTL<br>0  | CTL<br>1  | CTL<br>2       | CTL<br>3  | CTL<br>9C<br>9 | CTL<br>9 | CTL<br>86<br>8 | CTL<br>7F<br>7 | CTL<br>97<br>8 | CTL<br>8D<br>9 | CTL<br>8E<br>A | CTL<br>B<br>B  | CTL<br>C<br>C  | CTL<br>D<br>D  | CTL<br>E<br>E  | CTL<br>F<br>F  |
| 10 | CTL<br>10 | CTL<br>11 | CTL<br>12      | CTL<br>13 | CTL<br>9D<br>A | CTL<br>A | CTL<br>8<br>8  | CTL<br>87<br>7 | CTL<br>18<br>8 | CTL<br>19<br>9 | CTL<br>92<br>A | CTL<br>8F<br>B | CTL<br>1C<br>C | CTL<br>1D<br>D | CTL<br>1E<br>E | CTL<br>1F<br>F |
| 20 | CTL<br>80 | CTL<br>81 | CTL<br>1C<br>8 | CTL<br>83 | CTL<br>84<br>A | CTL<br>A | CTL<br>17<br>7 | CTL<br>1B<br>B | CTL<br>88<br>8 | CTL<br>89<br>9 | CTL<br>8A<br>A | CTL<br>8B<br>B | CTL<br>8C<br>C | CTL<br>5<br>5  | CTL<br>6<br>6  | CTL<br>7<br>7  |
| 30 | CTL<br>90 | CTL<br>91 | CTL<br>16<br>9 | CTL<br>93 | CTL<br>94<br>A | CTL<br>A | CTL<br>96<br>8 | CTL<br>4<br>4  | CTL<br>98<br>8 | CTL<br>99<br>9 | CTL<br>9A<br>A | CTL<br>9B<br>B | CTL<br>14<br>C | CTL<br>15<br>D | CTL<br>9E<br>E | CTL<br>1A<br>F |
| 40 | 20        | A0        | â<br>E2        | ä<br>E4   | à<br>E0        | á<br>E1  | ã<br>E3        | â<br>E5        | ç<br>E7        | ñ<br>F1        | ¢<br>A2        | .<br>2E        | <<br>3C        | (<br>28        | +<br>2B        | <br>7C         |
| 50 | &<br>26   | é<br>E9   | ê<br>EA        | ë<br>EB   | è<br>E8        | í<br>ED  | î<br>EE        | ï<br>EF        | ì<br>EC        | ß<br>DF        | !<br>21        | \$<br>24       | *<br>2A        | )<br>29        | ;<br>3B        | ^<br>5E        |
| 60 | -<br>2D   | /<br>2F   | Â<br>C2        | Ä<br>C4   | À<br>C0        | Á<br>C1  | Ã<br>C3        | Å<br>C5        | Ç<br>C7        | Ñ<br>D1        | ¡<br>A6        | ,<br>2C        | %<br>25        | _<br>5F        | ><br>3E        | ?<br>3F        |
| 70 | ø<br>F8   | É<br>C9   | Ê<br>CA        | Ë<br>CB   | È<br>C8        | Í<br>CD  | Î<br>CE        | Ï<br>CF        | Ì<br>CC        | ·<br>60        | :<br>3A        | #<br>23        | @<br>40        | '<br>27        | =<br>3D        | "<br>22        |
| 80 | Ø<br>D8   | a<br>61   | b<br>62        | c<br>63   | d<br>64        | e<br>65  | f<br>66        | g<br>67        | h<br>68        | i<br>69        | «<br>AB        | »<br>BB        | ð<br>F0        | ý<br>FD        | þ<br>FE        | ±<br>B1        |
| 90 | °<br>B0   | j<br>6A   | k<br>6B        | l<br>6C   | m<br>6D        | n<br>6E  | o<br>6F        | p<br>70        | q<br>71        | r<br>72        | ª<br>AA        | º<br>BA        | æ<br>E6        | ,<br>B8        | Æ<br>C6        | €<br>20AC      |
| A0 | µ<br>B5   | ~<br>7E   | s<br>73        | t<br>74   | u<br>75        | v<br>76  | w<br>77        | x<br>78        | y<br>79        | z<br>7A        | ı<br>A1        | ¿<br>BF        | Đ<br>D0        | [<br>5B        | Þ<br>DE        | ®<br>AE        |

# Characterset WE8ISO8859P1

|    | 0         | 1         | 2         | 3         | 4         | 5         | 6         | 7         | 8         | 9         | A         | B         | C         | D         | E         | F         |
|----|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 0  | CTL<br>0  | CTL<br>1  | CTL<br>2  | CTL<br>3  | CTL<br>4  | CTL<br>5  | CTL<br>6  | CTL<br>7  | CTL<br>8  | CTL<br>9  | CTL<br>A  | CTL<br>B  | CTL<br>C  | CTL<br>D  | CTL<br>E  | CTL<br>F  |
| 10 | CTL<br>10 | CTL<br>11 | CTL<br>12 | CTL<br>13 | CTL<br>14 | CTL<br>15 | CTL<br>16 | CTL<br>17 | CTL<br>18 | CTL<br>19 | CTL<br>1A | CTL<br>1B | CTL<br>1C | CTL<br>1D | CTL<br>1E | CTL<br>1F |
| 20 | 20        | !         | "         | #         | \$        | %         | &         | '         | (         | )         | *         | +         | ,         | -         | .         | /         |
| 30 | 0         | 1         | 2         | 3         | 4         | 5         | 6         | 7         | 8         | 9         | :         | ;         | <         | =         | >         | ?         |
| 40 | @         | A         | B         | C         | D         | E         | F         | G         | H         | I         | J         | K         | L         | M         | N         | O         |
| 50 | P         | Q         | R         | S         | T         | U         | V         | W         | X         | Y         | Z         | [         | \         | ]         | ^         | _         |
| 60 | `         | a         | b         | c         | d         | e         | f         | g         | h         | i         | j         | k         | l         | m         | n         | o         |
| 70 | p         | q         | r         | s         | t         | u         | v         | w         | x         | y         | z         | {         |           | }         | ~         | CTL       |
| 80 | UNK<br>80 | UNK<br>81 | UNK<br>82 | UNK<br>83 | UNK<br>84 | UNK<br>85 | UNK<br>86 | UNK<br>87 | UNK<br>88 | UNK<br>89 | UNK<br>8A | UNK<br>8B | UNK<br>8C | UNK<br>8D | UNK<br>8E | UNK<br>8F |
| 90 | UNK<br>90 | UNK<br>91 | UNK<br>92 | UNK<br>93 | UNK<br>94 | UNK<br>95 | UNK<br>96 | UNK<br>97 | UNK<br>98 | UNK<br>99 | UNK<br>9A | UNK<br>9B | UNK<br>9C | UNK<br>9D | UNK<br>9E | UNK<br>9F |
| A0 |           | ı         | ¢         | £         | ¤         | ¥         | ¦         | §         | ¨         | ©         | ª         | «         | ¬         |           | ®         | ¯         |
| B0 | °         | ±         | ²         | ³         | ´         | µ         | ¶         | ·         | ¸         | ¹         | º         | »         | ¼         | ½         | ¾         | ¿         |
| C0 | À         | Á         | Â         | Ã         | Ä         | Å         | Æ         | Ç         | È         | É         | Ê         | Ë         | Ì         | Í         | Î         | Ï         |



# Characterset WE8ISO8859P15

|    | 0          | 1          | 2          | 3          | 4          | 5          | 6          | 7          | 8          | 9          | A          | B          | C          | D          | E          | F          |
|----|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| 0  | CTRL<br>0  | CTRL<br>1  | CTRL<br>2  | CTRL<br>3  | CTRL<br>4  | CTRL<br>5  | CTRL<br>6  | CTRL<br>7  | CTRL<br>8  | CTRL<br>9  | CTRL<br>A  | CTRL<br>B  | CTRL<br>C  | CTRL<br>D  | CTRL<br>E  | CTRL<br>F  |
| 10 | CTRL<br>10 | CTRL<br>11 | CTRL<br>12 | CTRL<br>13 | CTRL<br>14 | CTRL<br>15 | CTRL<br>16 | CTRL<br>17 | CTRL<br>18 | CTRL<br>19 | CTRL<br>1A | CTRL<br>1B | CTRL<br>1C | CTRL<br>1D | CTRL<br>1E | CTRL<br>1F |
| 20 |            | !          | "          | #          | \$         | %          | &          | '          | (          | )          | *          | +          | ,          | -          | .          | /          |
|    | 20         | 21         | 22         | 23         | 24         | 25         | 26         | 27         | 28         | 29         | 2A         | 2B         | 2C         | 2D         | 2E         | 2F         |
| 30 | 0          | 1          | 2          | 3          | 4          | 5          | 6          | 7          | 8          | 9          | :          | ;          | <          | =          | >          | ?          |
|    | 30         | 31         | 32         | 33         | 34         | 35         | 36         | 37         | 38         | 39         | 3A         | 3B         | 3C         | 3D         | 3E         | 3F         |
| 40 | @          | A          | B          | C          | D          | E          | F          | G          | H          | I          | J          | K          | L          | M          | N          | O          |
|    | 40         | 41         | 42         | 43         | 44         | 45         | 46         | 47         | 48         | 49         | 4A         | 4B         | 4C         | 4D         | 4E         | 4F         |
| 50 | P          | Q          | R          | S          | T          | U          | V          | W          | X          | Y          | Z          | [          | \          | ]          | ^          | _          |
|    | 50         | 51         | 52         | 53         | 54         | 55         | 56         | 57         | 58         | 59         | 5A         | 5B         | 5C         | 5D         | 5E         | 5F         |
| 60 | `          | a          | b          | c          | d          | e          | f          | g          | h          | i          | j          | k          | l          | m          | n          | o          |
|    | 60         | 61         | 62         | 63         | 64         | 65         | 66         | 67         | 68         | 69         | 6A         | 6B         | 6C         | 6D         | 6E         | 6F         |
| 70 | p          | q          | r          | s          | t          | u          | v          | w          | x          | y          | z          | {          |            | }          | ~          | CTRL       |
|    | 70         | 71         | 72         | 73         | 74         | 75         | 76         | 77         | 78         | 79         | 7A         | 7B         | 7C         | 7D         | 7E         | 7F         |
| 80 |            |            |            |            |            |            |            |            |            |            |            |            |            |            |            |            |
| 90 |            |            |            |            |            |            |            |            |            |            |            |            |            |            |            |            |
| A0 |            | ı          | ç          | £          | €          | ¥          | Š          | Ş          | š          | ©          | ª          | «          | ¬          |            | ®          | —          |
|    | A0         | A1         | A2         | A3         | 20AC       | A6         | 160        | A7         | 161        | A9         | AA         | AB         | AC         | AD         | AE         | AF         |
| B0 | °          | ±          | ²          | ³          | Ž          | µ          | ¶          | ·          | ž          | ı          | º          | »          | Œ          | œ          | ÿ          | ¿          |
|    | B0         | B1         | B2         | B3         | 17D        | B5         | B6         | B7         | 17E        | B9         | BA         | BB         | 152        | 153        | 178        | BF         |
|    | À          | Á          | Â          | Ã          | Ä          | Å          | Æ          | Ç          | È          | É          | Ê          | Ë          | Ì          | Í          | Î          | Ï          |

# Charactersets

- In the previous example the character set WE8ISO8859P15 is the correct choice
- Character set on the Oracle for Linux on z platform  
=  
Character set of the source database ?

# Agenda

- Linux on System z configuration variations
- Load profile determination of the source environment
- Characterset differences
- Schema and data migration
- Impact on applications
- Best practices

# Migrating data with its definitions

- Source database on z/OS
  - Datapump utility and export utility are available to move data with its definitions, but datapump uses USS
  - Recommendation: Use the export utility on z/OS

# Migrating data with its definitions

- Restrictions of export/import
  - Java classes, resources and procedures that are created using EJB's are not processed
  - CONSTRAINTS with RELY attribute are not processed
  - Table compression is not performed during import
  - BFILES have to be copied to the target system manually
  - BINARY\_FLOAT and BINARY\_DOUBLE are not supported

# Export / Import Modes

- FULL
- TABLESPACE
- USER (Schema)
- TABLE

# Migrating data with its definitions

- Datapump Utility

**Data Pump Export** is a utility for unloading data and metadata into a set of operating system files called a **dump file set**. The dump file set can be imported only by the Data Pump Import utility. The dump file set can be imported on the same system or it can be moved to another system and

**Data Pump Import** is a utility for loading an export dump file set into a **target system**. The dump file set is made up of one or more disk files that contain table data, database object metadata, and control information. The files are written in a proprietary, binary format. During an import operation, the Data Pump Import utility uses these files to locate each database object in the dump file set

Source: Oracle® Database Utilities

# Migrating data with its definitions

- Datapump Utility advantages
  - Datapump operations are executed in the Oracle database server.
  - Datapump export and import use parallel execution rather than a single stream of execution, for improved performance.
  - Export and import jobs can be stopped and re-started

# Migrating data with its definitions

- Datapump Utility advantages
  - Detach and re-attach to long running Jobs
  - If a table has compression enabled, Data Pump Import attempts to compress the data being loaded.
  - Space estimation

# Datapump Export / Import Modes

- FULL
- SCHEMA
- TABLE
- TABLESPACE
- Transportable Tablespace

# Migrating data with its definitions

- Transportable Tablespaces
  - The source and the Linux for z database must use the same character set
  - A tablespace with the same name may not exist on the Oracle for Linux on system z target database
  - Objects which are external to tablespaces (e.g. PL/SQL procedures) have to be migrated separately
  - System, undo, sysaux and temp tablespaces cannot be transported

# Migrating data with its definitions

- Transportable Tablespaces
  - Have to be self contained

# Migrating data with its definitions

- Determine endianness of the source and target platform

```
SQL> select * from v$transportable_platform order by platform_id;
```

| PLATFORM_ID | PLATFORM_NAME                 | ENDIAN_FORMAT |
|-------------|-------------------------------|---------------|
| 1           | Solaris[tm] OE (32-bit)       | Big           |
| 2           | Solaris[tm] OE (64-bit)       | Big           |
| 3           | HP-UX (64-bit)                | Big           |
| 4           | HP-UX IA (64-bit)             | Big           |
| 5           | HP Tru64 UNIX                 | Little        |
| 6           | AIX-Based Systems (64-bit)    | Big           |
| 7           | Microsoft Windows IA (32-bit) | Little        |
| 8           | Microsoft Windows IA (64-bit) | Little        |
| 9           | IBM zSeries Based Linux       | Big           |

# Migrating data with its definitions

- Transportable Tablespaces - example
  - Source System: Oracle on Linux x86
    - Little endian platform
  - Target System Oracle on Linux for z
    - Big endian platform
  - ASM is not used

# Migrating data with its definitions

- Transportable Tablespaces – example
  - Build directories and grant authorities on the source and the target system. Those directories are used to hold:
    - Tablespace metadata
    - Export logfile

```
create directory tniewel_dmp as '/home/oracle/tniewel/tts/';  
create directory tts_dump_log as '/home/oracle/tniewel/tts/';
```

# Migrating data with its definitions

- Transportable Tablespaces – example (Source)
  - Set all tablespaces which have to be transported to Read Only(RO)
  - The tablespaces have to be self contained

```
ALTER TABLESPACE tniewel1 READ ONLY;
ALTER TABLESPACE tniewel2 READ ONLY;
exec DBMS_TTS.TRANSPORT_SET_CHECK('tniewel1,tniewel2',TRUE);
SELECT * FROM TRANSPORT_SET_VIOLATIONS;
```

# Migrating data with its definitions

- Transportable Tablespaces – example (Source)
  - Export the metadata into the defined directory

```
expdp system/oracle directory=tniewel_dmp dumpfile=tn1.dmp  
logfile=tts_dump_log:tts.log  
transport_tablespaces=tniewel1,tniewel2  
transport_full_check=y
```

# Migrating data with its definitions

- Transportable Tablespaces – example (Source)
  - Convert the endianness with RMAN

```
RMAN> convert tablespace tniewel1 to platform 'IBM zSeries
based Linux' format '/home/oracle/tniewel/tts/tniewel_1.dbf';
RMAN> convert tablespace tniewel2 to platform 'IBM zSeries
Based Linux' format '/home/oracle/tniewel/tts/tniewel_2.dbf';
```

```
Starting backup at 22-FEB-10 using channel ORA_DISK_1
channel ORA_DISK_1: starting datafile conversion
Input datafile fno=00007 name=/space/oradata/ora10gR2/ora10gR2/tniewel_2.dbf
converted datafile=/home/oracle/tniewel/tts/tniewel_2.dbf
channel ORA_DISK_1: datafile conversion complete, elapsed time: 00:00:01
Finished backup at 22-FEB-10
```

# Migrating data with its definitions

- Transportable Tablespaces – example (Target)
  - Alternative: Convert the endianness with RMAN on the Linux for system z platform

```
RMAN>convert datafile '/home/oracle/tniewel/tniewel_1.dbf' FROM  
PLATFORM 'Linux IA (32-bit) '
```

```
Starting backup at 22-FEB-10  
using channel ORA_DISK_1  
channel ORA_DISK_1: starting datafile conversion  
Input filename=/home/oracle/tniewel/tniewel_1.dbf  
converted datafile=/oracle/oracledb/dbs/data_D-ORA10GR2_I-3971714815_TS-TNIEWEL1_FNO-  
6_0116krdp  
channel ORA_DISK_1: datafile conversion complete, elapsed time: 00:00:01  
Finished backup at 22-FEB-10
```

# Migrating data with its definitions

- Transportable Tablespaces – example (Target)
  - Copy the dump file and the datafiles to the Linux for z target machine
  - If ASM is used, DBMS\_FILE\_TRANSFER has to be used

# Migrating data with its definitions

- Transportable Tablespaces – example (Target)
  - Tailor the datapump import parameter file

```
directory=tniewel_dmp
  dumpfile=tn1.dmp
  logfile=tts1_imp.log
  keep_master=y
TRANSPORT_DATAFILES='/home/oracle/tniewel/tniewel_1.dbf',
'/home/oracle/tniewel/tniewel_2.dbf'
```

# Migrating data with its definitions

- Transportable Tablespaces – example (Target)

- Import Metadata

```
impdp system/manager parfile=imp.par
```

```
Import: Release 10.2.0.4.0 - 64bit Production on Monday, 22 February, 2010 4:24:59
```

```
Copyright (c) 2003, 2007, Oracle. All rights reserved.
```

```
Connected to: Oracle Database 10g Enterprise Edition Release 10.2.0.4.0 - 64bit  
Production
```

```
With the Partitioning, OLAP, Data Mining and Real Application Testing options
```

```
Master table "SYSTEM"."SYS_IMPORT_TRANSPORTABLE_01" successfully loaded/unloaded
```

```
Starting "SYSTEM"."SYS_IMPORT_TRANSPORTABLE_01": system/***** parfile=imp.par
```

```
Processing object type TRANSPORTABLE_EXPORT/PLUGTS_BLK
```

```
Processing object type TRANSPORTABLE_EXPORT/TABLE
```

```
Processing object type TRANSPORTABLE_EXPORT/POST_INSTANCE/PLUGTS_BLK
```

```
Job "SYSTEM"."SYS_IMPORT_TRANSPORTABLE_01" successfully completed at 04:25:09
```

# Migrating data with its definitions

- Transportable Tablespaces – example (Target)
  - Set tablespaces Read Write

```
ALTER TABLESPACE tniewel1 READ write;  
ALTER TABLESPACE tniewel2 READ write;
```

# Migrating data with its definitions

- Transportable Database
  - Automates the movement of a complete database to another platform
  - The source and destination platform must have the same endianness

# Agenda

- Linux on System z configuration variations
- Load profile determination of the source environment
- Characterset differences
- Schema and data migration
- **Impact on applications**
- Best practices

# Impact on applications

- If the architecture changes from local client access to remote client access
  - Use bulk operations to minimize the network latency
  - Precompiler Options
    - HOLD\_CURSOR=YES
    - RELEASE\_CURSOR=NO
    - PREFETCH=nnn

# Impact on applications

- Network latency
  - XM : < 50ns
  - LAN: < 1ms
  - WAN: < 10ms - 500ms
- Program design
  - A large number of atomic operations results in a large number of network roundtrips.  
Working with result sets improves performance

# Impact on applications – Migrating Oracle for z/OS to z/Linux

- z/OS is an EBCDIC Platform
- z/Linux is an ASCII Platforms
- Applications may show a different behavior
  - Sort Order
    - Sort sequence of Order By and Group By clauses may be different
  - Ranges in where condition

# Sort Order

- Example
  - Create table tn1(f1 char(1));
  - select f1 from tn1 order by 1;
- z/OS
  - \$, #, A, B,C,D,E,F,G,H,0,1,2,3,4,5,6,7
- ASCII
  - #,\$,0,1,2,3,4,5,6,7 A, B,C,D,E,F,G,H
- Unicode
  - #,\$,0,1,2,3,4,5,6,7 A, B,C,D,E,F,G,H

## Between Predicate

- Create table tn1(f1 char(1));  
select \* from tn1 where f1 between '\$' and '5'  
order by 1;
- z/OS
  - \$, #, A, B,C,D,E,F,G,H,0,1,2,3,4,5
- ASCII
  - \$,0,1,2,3,4,5
- Unicode
  - \$,0,1,2,3,4,5

# Solutions

- Change of the SQL Query
  - `select * from tn1 order by nlssort(F1, 'nls_sort=ebcdic');`
- Change of the NLS settings

# Solutions

- NLS parameter settings
  - NLS\_SORT
    - alter session set nls\_sort=ebcdic;
  - NLS\_COMP
    - alter session set nls\_comp=linguistic

# Solutions

- NLS\_SORT

**NLS\_SORT specifies the type of sort for character data.** It overrides the default value that is derived from NLS\_LANGUAGE. Note: When the NLS\_SORT parameter is set to BINARY, the optimizer can, in some cases, satisfy the ORDER BY clause without doing a sort by choosing an index scan. When NLS\_SORT is set to a linguistic sort, a sort is needed to satisfy the ORDER BY clause if there is no linguistic index for the linguistic sort specified by NLS\_SORT. If a linguistic index exists for the linguistic sort specified by NLS\_SORT, then the optimizer can, in some cases, satisfy the ORDER BY clause without doing a sort by choosing an index scan.

Source: Oracle® Database Globalization Support Guide

# Solutions

- NLS\_COMP

**The value of NLS\_COMP affects the comparison behavior of SQL operations.** You can use NLS\_COMP to avoid the cumbersome process of using the NLSSORT function in SQL statements when you want to perform a linguistic comparison instead of a binary comparison. When NLS\_COMP is set to LINGUISTIC, SQL operations perform a linguistic comparison based on the value of NLS\_SORT. A setting of ANSI is for backward compatibility; in general, you should set NLS\_COMP to LINGUISTIC when you want to perform a linguistic comparison.

Source: Oracle® Database Globalization Support Guide

# Solutions

- Linguistic indexes

**Linguistic sorting is language-specific and requires more data processing than binary sorting.** Using a binary sort for ASCII is accurate and fast because the binary codes for ASCII characters reflect their linguistic order. When data in multiple languages is stored in the database, you may want applications to sort the data returned from a `SELECT...ORDER BY` statement according to different sort sequences depending on the language. You can accomplish this without sacrificing performance by using linguistic indexes.

Source: Oracle® Database Globalization Support Guide

# Solutions

- Access Path with standard NLS settings and Index on column F1

```
select f1 from tnl where f1 between '4' and '5' order by 1;
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

| OPERATION         | OBJECT_NAME | OPTIONS    |
|-------------------|-------------|------------|
| SELECT STATEMENT  |             |            |
| INDEX             | I1          | RANGE SCAN |
| Access Predicates |             |            |
| AND               |             |            |
| F1 >= '4'         |             |            |
| F1 <= '5'         |             |            |

# Solutions

- Access Path with NLS\_COMP=LINGUISTIC and NLS\_SORT=EBCDIC

```
alter session set nls_sort=ebcdic;

alter session set nls_comp=linguistic;

select fl from tnl where fl between '4' and '5' order by 1;

Select * from tnl order by 1;
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

| OPERATION               | OBJECT_NAME | OPTIONS  | COST |
|-------------------------|-------------|----------|------|
| SELECT STATEMENT        |             |          | 4    |
| SORT                    |             | ORDER BY | 4    |
| TABLE ACCESS            | TN1         | FULL     | 3    |
| Filter Predicates       |             |          |      |
| AND                     |             |          |      |
| NLSSORT(INTERNAL_FUNCIO |             |          |      |
| NLSSORT(INTERNAL_FUNCIO |             |          |      |

# Solutions

- Access Path with NLS\_COMP=LINGUISTIC and NLS\_SORT=EBCDIC and Linguistic Index

```
alter session set nls_comp=linguistic;  
drop index il;  
CREATE INDEX il ON tnl (NLSSORT(f1, 'NLS_SORT = EBCDIC'));  
select f1 from tnl where f1 between '4' and '5' order by 1;
```

| OPERATION               | OBJECT_NAME | OPTIONS        | COST |
|-------------------------|-------------|----------------|------|
| SELECT STATEMENT        |             |                | 2    |
| TABLE ACCESS            | TN1         | BY INDEX ROWID | 2    |
| INDEX                   | I1          | RANGE SCAN     | 1    |
| Access Predicates       |             |                |      |
| AND                     |             |                |      |
| NLSSORT(INTERNAL_FUNCIO |             |                |      |
| NLSSORT(INTERNAL_FUNCIO |             |                |      |

# Agenda

- Linux on System z configuration variations
- Load profile determination of the source environment
- Characterset differences
- Schema and data migration
- Impact on applications
- **Best practices**

# Best practices

- Simulate the production workload
- Check performance characteristics of the source and target environment with AWR reports (e.g. awrddrpt.sql)
- Use AWR reports to drill down to queries with changed performance characteristics
- SQL Tuning Advisor(DBMS\_SQLTUNE) allows to capture queries on the source environment and to transport and evaluate the queries on Oracle for Linux on z environment

# Best practices

- SQL Tuning Sets
  - Can be generated out of AWR reports
  - Can be generated out of the cursor cache

```
BEGIN  DBMS_SQLTUNE.create_sqlset
        (sqlset_name  => 'test6_sql_tuning_set',
         description  => 'IOUG Tuning Set');
END;
```

# Best practices

```
BEGIN  OPEN l_cursor FOR
        SELECT VALUE(p)      FROM    TABLE (DBMS_SQLTUNE.select_workload_repository (
        10594, -- begin_snap
        10595, -- end_snap
        'executions >= 1000',
        NULL, -- object_filter
        NULL, -- ranking_measure1
        NULL, -- ranking_measure2
        NULL, -- ranking_measure3
        NULL, -- result_percentage
        1000) -- result_limit
    ) p;
        DBMS_SQLTUNE.load_sqlset (sqlset_name      => 'test6_sql_tuning_set',
        populate_cursor => l_cursor);
END;
```

# Best practices

- The `DBMS_SQLTUNE.CREATE_STGTAB_SQLSET` procedure creates a staging table

```
BEGIN
DBMS_SQLTUNE.CREATE_STGTAB_SQLSET( schema_name      =>'TNIEWEL',
                                   table_name        =>'STGTAB_SQLSET');
END;
```

- The `DBMS_SQLTUNE.PACK_STGTAB_SQLSET` moves the captured queries to the staging table

```
BEGIN
DBMS_SQLTUNE.PACK_STGTAB_SQLSET(sqlset_name      =>'test6_sql_tuning_set' ,
                                 staging_schema_owner =>'TNIEWEL',
                                 staging_table_name  =>'STGTAB_SQLSET');
END;
```

# Best practices

- Use datapump or database links to transfer the staging table to the Oracle for Linux on z system

```
expdp tniewel/tniewel directory=tniewel_dmp  
dumpfile=stgtab.dmp logfile=tn1.log tables=stgtab_sqlset
```

```
impdp tniewel/tniewel dumpfile=stgtab.dmp  
directory=tniewel_dmp
```

# Best practices

- The DBMS\_SQLTUNE.UNPACK\_STGTAB\_SQLSET procedure reads the data from the staging table

```
BEGIN
DBMS_SQLTUNE.UNPACK_STGTAB_SQLSET(sqlset_name           => '%',
                                   sqlset_owner          => '%',
                                   replace               => FALSE,
                                   staging_schema_owner  => 'TNIEWEL',
                                   staging_table_name    => 'STGTAB_SQLSET');

END;
```

- The SQL Statements in the SQL tuning set can be analyzed with Oracle Enterprise Manager or the DBMS\_SQLTUNE package

# Summary

- The migration to an Oracle for Linux on z database is similar to the migration to other Oracle platforms
  - The Linux on z environment has to be selected
  - A proper sizing of the target environment has to be carried out
  - The character set of the target environment has to be selected
  - Data and data definitions have to be transported to the target environment
  - Performance of the source and the target environment has to be compared



ORA