

# Database Backup Formatter Tool User Guide

DBN

USER GUIDE

**Copyright**

© Ericsson AB 2015, 2017, 2018. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

**Disclaimer**

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.



# Contents

<b>1</b>	<b>Overview</b>	<b>1</b>
1.1	Installation	1
1.2	Backup Path	1
1.3	Backup Directory	1
<b>2</b>	<b>BackupFormatter Command</b>	<b>3</b>
2.1	list_classes	3
2.2	dump_class	4
2.3	list_instances	4
2.4	dump_instance	5
2.5	dump_mdp_instances	6
2.6	dump_class_instances	8
<b>3</b>	<b>DBN Backup Formatter Tool Format</b>	<b>11</b>
3.1	Primitive Types	11
3.2	External Types	11
3.3	Composed Values	11
	<b>Reference List</b>	<b>13</b>





# 1 Overview

DBN in-memory database objects can be backed up to disk through DBN backup. The DBN Backup Formatter Tool exports information from a DBN backup (binary format) and prints it in a human readable format. This User Guide describes how to use the Backup Formatter Tool functionality.

DBN backups are incorporated into BRF backups. For information on how to create and export a BRF backup, refer to BRF-C Management Guide, Reference [1].

## 1.1 Installation

Normally, the application provides the Database Formatter Tool in its packages. Therefore, refer to the application documentation for more information on the installation of the tool.

If the application does not include the tool by default, then it is possible to install it by doing these steps:

1. Download the required version of the following package:

**DBS RT: CXP 902 5264/5**

2. Extract the package and get the Backup Formatter Tool from the following path:

```
delivery\PSR-DBS_RT-CXP9025264-<DBS_version>\contained\PSR-DBS_RT-CXP9030894-<DBS_version>\contained\TDP-DbTools-CXP9040378-<DBS_version>\contained\DT_BackupFormatter\share\BackupFormatter
```

3. The tool can be executed directly on the node or on any 64-bit host machine. Run the binary without parameters to get a help about available options. For more information on the available command options, see Section 2 on page 3.

## 1.2 Backup Path

The backups are stored at the following path:

```
/storage/system/config/dbsv/backup/
```

The backup directories use the following naming convention: <YYYYMMDD>.<HHMMSS>.<microseconds>, for example, 20171130.143437.262994.

## 1.3 Backup Directory

A backup directory contains the following files:



mdps.PL-3.data.idx	mdps.PL-4.data.idx	mdps.PL-5.data.idx	mdps.SC-1.data.idx	mdps.SC-2.data.idx
mdps.PL-3.data	mdps.PL-4.data	mdps.PL-5.data	mdps.SC-1.data	mdps.SC-2.data
schema.gz				

#### Example 1 Backup Directory Contents Example

The .data files contain the primary data replicas.

The .idx files contain the position of the Master Data Partitions (MDPs) within the .data file of the same name.



## 2 BackupFormatter Command

The command has the following syntax:

**BackupFormatter** <backup\_directory> <command> [<arguments>]

Where the <backup\_directory> refers to this directory:

/storage/system/config/dbsv/backup/<backup\_name>/databaseBackup/

The options for <command> are shown in Table 1.

Table 1 Command Options for BackupFormatter

Option	Description
list_classes	Lists all classes (POTs) with their IDs and names.
dump_class	Dumps information about the given classes (POTs).
list_instances	Lists all records (POT instances) of the given classes.
dump_instance	Dumps information about the given records (POT instances).
dump_mdp_instances	Dumps the records (POT instances) in the given MDP.
dump_class_instances	Dumps the records (POT instances) in the MDPs of the given class (POT).

### 2.1 list\_classes

This command option lists all classes with their IDs and names.

The number in the list indicates the class ID and the name after it indicates the class name.

**Note:** The class ID is also referred to as POT Runtime ID (RTID) or POT Class Number (Class No).

#### Example

```
>$ BackupFormatter <backup_directory> list_classes
```

#### Output



```
class ID class name
1117352 SubscriberTypeA
1117358 SubscriberTypeB
1117366 SubscriberType1
1117372 SubscriberType2
```

## 2.2 dump\_class

This command dumps the class attributes: the name and the type. For more information on the printout values, see Table 2 and Table 3.

Both the class ID or the class name can be used as the command option argument.

### Example

```
>$ BackupFormatter <backup_directory> dump_class 1117352
```

Or

```
>$ BackupFormatter <backup_directory> dump_class SubscriberTypeA
```

### Output

```
1117352 SubscriberTypeA
0. subKey:U
1. updateAtt:U
2. array:A0[:U]
3. servObj:G
4. checksumAttr:I
5. groupSize:I
```

## 2.3 list\_instances

This command lists all records of the given classes.

The records are listed in the following format:

```
<MDP_no> -<seq_no> [key_type:key_value]
```

Where:

- <MDP\_no> is the MDP number calculated as:  $RTID \times 2^{32} + DU \times 2^{16}$ .
- <seq\_no> is the sequence number of the record within the MDP, in hexadecimal format.
- key\_type is the key type (see Table 2).
- key\_value is the value of the key.

### Example

```
>$ BackupFormatter <backup_directory> list_instances 1117352
```





**Note:** The command option only accepts primary key defining classes as a command parameter. Do not use unkeyed classes or classes derived from keyed classes.

### Output

```
===== Instances of class 1117352 =====
4798990298120192-0000380000000003 [U:0]
4798990298185728-0000480000000003 [U:1]
4798990298251264-0000580000000003 [U:2]
4798990298316800-0000180000000003 [U:3]
4798990298382336-0000280000000003 [U:4]
4798990298447872-0000380000000003 [U:5]
4798990298513408-0000480000000003 [U:6]
4798990298578944-0000580000000003 [U:7]
4798990298644480-0000180000000003 [U:8]
4798990298710016-0000280000000003 [U:9]
```

## 2.4 dump\_instance

This command lists the entire content of a record, including attribute names, types, and values. The attribute values are listed both in plain text (where “~” replaces all non-readable characters) and in hexadecimal format.

### Example

```
>$ BackupFormatter <backup_directory> dump_instance
4798990298447872-0000380000000003
```

**Note:** The command option only accepts primary key defining classes as a command parameter. Do not use unkeyed classes or classes derived from keyed classes.

### Output

[illegible]

## 2.5 dump mdp instances

This command works similarly to the `dump_instance` command option (see Section 2.4 on page 5), but it prints the content of all records belonging to the given MDP number. To list the MDP numbers of a class, use the `list_instances` command option, see Section 2.3 on page 4.





## 2.6 dump\_class\_instances

This command works similarly to the `dump_instance` command option (see Section 2.4 on page 5), but instead of listing the entire content of a specific record, it lists the entire content of all records that belong to that class.

These class types are identified either by their class number or their names. The command lists the content of all records where the class type matches the class number or name.

### Example

```
>$ BackupFormatter <backup_directory> dump_class_instances 1117352
```

### Output

9





## 3 DBN Backup Formatter Tool Format

The Backup Formatter Tool uses short notations to represent attribute types in classes or records. The notations are described in the following subsections.

### 3.1 Primitive Types

Table 2 shows the primitive types.

Table 2 Primitive Types

Integer	I
Unsigned	U
Unsigned long	L
Long	K
Double	D
Char	C
Bool	B
Octet	O
String	S

### 3.2 External Types

Any external type is stored as an octet array.

### 3.3 Composed Values

Table 3 shows the composed values.

Table 3 Composed Values

Reference (DID)	G
Array	A<size>[:T] <sup>(1)(2)</sup>
Multiref	M
Record	R[:T:T:T:T... :T] <sup>(2)</sup>

(1) The size is zero for dynamic arrays.

(2) T can be any type using the same notations, for example: A0[:G] is a dynamic array of references (DIDs).







## Reference List

### Documents

- [1] BRF-C Management Guide, 1/1553-APR 901 0485/1