Rocky Enterprise Linux 9.2 Manual Pages on command 'ibd2sdi.1'

$ man ibd2sdi.1

IBD2SDI(1) MySQL Database System IBD2SDI(1)

NAME

ibd2sdi - InnoDB utility for extracting serialized dictionary information (SDI) from an InnoDB tablespace

SYNOPSIS

ibd2sdi [options] file_name1 [file_name2 file_name3 ...]

DESCRIPTION

ibd2sdi is a utility for extracting serialized dictionary information (SDI) from InnoDB tablespace files. SDI data is present in all persistent InnoDB tablespace files.

ibd2sdi can be run on file-per-table tablespace files (*.ibd files), general tablespace files (*.ibd files), system tablespace files (ibdata* files), and the data dictionary tablespace (mysql.ibd). It is not supported for use with temporary tablespaces or undo tablespaces.

ibd2sdi can be used at runtime or while the server is offline. During DDL operations, ROLLBACK operations, and undo log purge operations related to SDI, there may be a short interval of time when ibd2sdi fails to read SDI data stored in the tablespace.

ibd2sdi performs an uncommitted read of SDI from the specified

tablespace. Redo logs and undo logs are not accessed.

Invoke the ibd2sdi utility like this:

```
ibd2sdi [options] file_name1 [file_name2 file_name3 ...]
```

ibd2sdi supports multi-file tablespaces like the InnoDB system

tablespace, but it cannot be run on more than one tablespace at a time.

For multi-file tablespaces, specify each file:

```
ibd2sdi ibdata1 ibdata2
```

The files of a multi-file tablespace must be specified in order of the

ascending page number. If two successive files have the same space ID,

the later file must start with the last page number of the previous

file + 1.

ibd2sdi outputs SDI (containing id, type, and data fields) in JSON

format. ibd2sdi Options

ibd2sdi supports the following options:

? --help, -h Display a help message and exit. For example:

```
Usage: ./ibd2sdi [-v] [-c <strict-check>] [-d <dump file name>] [-n] filename1 [filenames]
See http://dev.mysql.com/doc/refman/8.0/en/ibd2sdi.html for usage hints.
  -h, --help          Display this help and exit.
  -v, --version       Display version information and exit.
  -#, --debug[=name]  Output debug log. See
                      http://dev.mysql.com/doc/refman/8.0/en/dbug-package.html
  -d, --dump-file=name
                      Dump the tablespace SDI into the file passed by user.
                      Without the filename, it will default to stdout
  -s, --skip-data     Skip retrieving data from SDI records. Retrieve only id
                      and type.
  -i, --id=#          Retrieve the SDI record matching the id passed by user.
  -t, --type=#        Retrieve the SDI records matching the type passed by
                      user.
  -c, --strict-check=name
                      Specify the strict checksum algorithm by the user.
                      Allowed values are innodb, crc32, none.
```

-n, --no-check Ignore the checksum verification.

-p, --pretty Pretty format the SDI output.If false, SDI would be not human readable but it will be of less size (Defaults to on; use --skip-pretty to disable.)

Variables (--variable-name=value)

and boolean options {FALSE|TRUE} Value (after reading options)

--------------------------------- --

debug (No default value)

dump-file (No default value)

skip-data FALSE

id 0

type 0

strict-check crc32

no-check FALSE

pretty TRUE

? --version, -v Display version information and exit. For example:

ibd2sdi Ver 8.0.3-dmr for Linux on x86_64 (Source distribution)

? --debug[=debug_options], -# [debug_options] Prints a debug log. For debug options, refer to Section 5.9.4, ?The DBUG Package?.

ibd2sdi --debug=d:t /tmp/ibd2sdi.trace

This option is available only if MySQL was built using WITH_DEBUG. MySQL release binaries provided by Oracle are not built using this option.

? --dump-file=, -d Dumps serialized dictionary information (SDI) into the specified dump file. If a dump file is not specified, the tablespace SDI is dumped to stdout.

ibd2sdi --dump-file=file_name ../data/test/t1.ibd

? --skip-data, -s Skips retrieval of data field values from the serialized dictionary information (SDI) and only retrieves the id and type field values, which are primary keys for SDI records.

$> ibd2sdi --skip-data ../data/test/t1.ibd

["ibd2sdi"

,

{

"type": 1,

"id": 330

}

,

{

"type": 2,

"id": 7

}

]

? --id=#, -i # Retrieves serialized dictionary information (SDI) matching the specified table or tablespace object id. An object id is unique to the object type. Table and tablespace object IDs are also found in the id column of the mysql.tables and mysql.tablespace data dictionary tables. For information about data dictionary tables, see Section 14.1, ?Data Dictionary Schema?.

$> ibd2sdi --id=7 ../data/test/t1.ibd

["ibd2sdi"

,

{

"type": 2,

"id": 7,

"object":

{

"mysqld_version_id": 80003,

"dd_version": 80003,

"sdi_version": 1,

"dd_object_type": "Tablespace",

"dd_object": {

"name": "test/t1",

"comment": "",

"options": "",

"se_private_data": "flags=16417;id=2;server_version=80003;space_version=1;",

```
      "engine": "InnoDB",
      "files": [
        {
          "ordinal_position": 1,
          "filename": "./test/t1.ibd",
          "se_private_data": "id=2;"
        }
      ]
    }
  }
  }
  ]
```


? --type=#, -t # Retrieves serialized dictionary information (SDI) matching the specified object type. SDI is provided for table (type=1) and tablespace (type=2) objects.

This example show output for a tablespace ts1 in the test database:

```
$> ibd2sdi --type=2 ../data/test/ts1.ibd
["ibd2sdi"
,
{
   "type": 2,
   "id": 7,
   "object":
      {
    "mysqld_version_id": 80003,
    "dd_version": 80003,
    "sdi_version": 1,
    "dd_object_type": "Tablespace",
    "dd_object": {
      "name": "test/ts1",
      "comment": "",
      "options": "",
      "se_private_data": "flags=16417;id=2;server_version=80003;space_version=1;",
```

```
            "engine": "InnoDB",
            "files": [
                {
                    "ordinal_position": 1,
                    "filename": "./test/ts1.ibd",
                    "se_private_data": "id=2;"
                }
            ]
        }
    }
    }
    ]
```


Due to the way in which InnoDB handles default value metadata, a default value may be present and non-empty in ibd2sdi output for a given table column even if it is not defined using DEFAULT.

Consider the two tables created using the following statements, in the database named i:

```
CREATE TABLE t1 (c VARCHAR(16) NOT NULL);
CREATE TABLE t2 (c VARCHAR(16) NOT NULL DEFAULT "Sakila");
```

Using ibd2sdi, we can see that the default_value for column c is nonempty and is in fact padded to length in both tables, like this:

```
$> ibd2sdi ../data/i/t1.ibd  | grep -m1 '\"default_value\"' | cut -b34- | sed -e s/,//
```


"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA\nAAAA
AAAAAAA="


```
$> ibd2sdi ../data/i/t2.ibd  | grep -m1 '\"default_value\"' | cut -b34- | sed -e s/,//
```


"BlNha2lsYQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA\nAAAAAA
AAAAA="

Examination of ibd2sdi output may be easier using a JSON-aware utility like jq[1], as shown here:

```
$> ibd2sdi ../data/i/t1.ibd  | jq '.[1]["object"]["dd_object"]["columns"][0]["default_value"]'
```

"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA\nAAAA
AAAAAAA="

```
$> ibd2sdi ../data/i/t2.ibd  | jq '.[1]["object"]["dd_object"]["columns"][0]["default_value"]'
```

"BlNha2lsYQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA\nAAAAAA
AAAAA="

For more information, see the MySQL Internals documentation[2].

? --strict-check, -c Specifies a strict checksum algorithm for validating the checksum of pages that are read. Options include innodb, crc32, and none.

In this example, the strict version of the innodb checksum algorithm is specified:

```
ibd2sdi --strict-check=innodb ../data/test/t1.ibd
```

In this example, the strict version of crc32 checksum algorithm is specified:

```
ibd2sdi -c crc32 ../data/test/t1.ibd
```

If you do not specify the --strict-check option, validation is performed against non-strict innodb, crc32 and none checksums.

? --no-check, -n Skips checksum validation for pages that are read.

```
ibd2sdi --no-check ../data/test/t1.ibd
```

? --pretty, -p Outputs SDI data in JSON pretty print format. Enabled by default. If disabled, SDI is not human readable but is smaller in size. Use --skip-pretty to disable.

```
ibd2sdi --skip-pretty ../data/test/t1.ibd
```

COPYRIGHT

You should have received a copy of the GNU General Public License along with the program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA or see http://www.gnu.org/licenses/.

NOTES

1. jq

   https://stedolan.github.io/jq/

2. MySQL Internals documentation

   https://dev.mysql.com/doc/dev/

SEE ALSO

For more information, please refer to the MySQL Reference Manual, which may already be installed locally and which is also available online at http://dev.mysql.com/doc/.

AUTHOR

Oracle Corporation (http://dev.mysql.com/).