Hash Code Identity in the Database

Jaromír D.B. Nemec

UKOUG 2018

Who am I

15 years Oracle experience

10g Oracle Certified Associate

both OLTP and DWH

Oracle conference speaker

Oracle Magazine Peer 09/2006

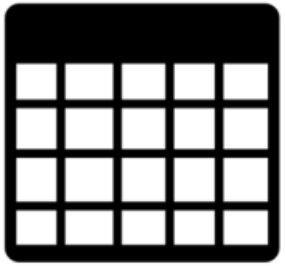
Oracle Beta-Tester

NoSQL, Big Data

Machine Learning

Why Hash Code?

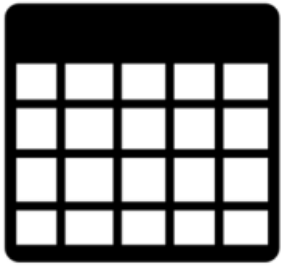
A



?

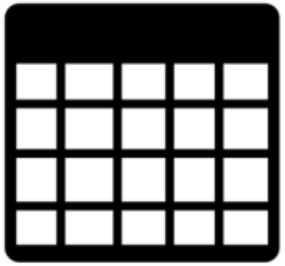
=

B



Why Hash Code?

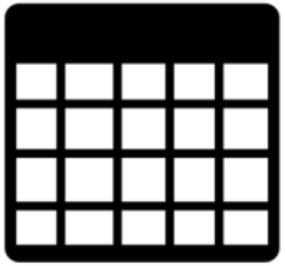
A



?

=

B

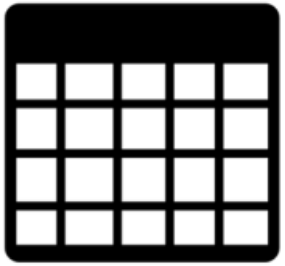


select count(*) from A;

COUNT(*)
3

Why Hash Code?

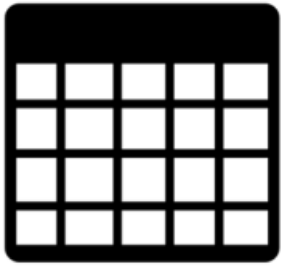
A



?

=

B



select count(*) from A;

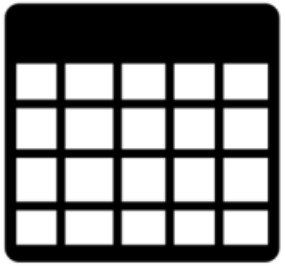
COUNT(*)
3

select count(*) from B;

COUNT(*)
3

(A – B) + (B – A)

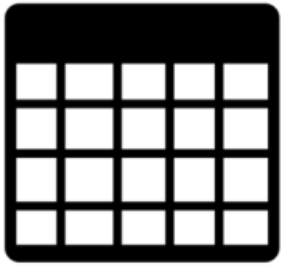
A



?

=

B



select col from A

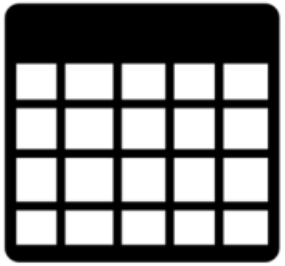
minus

select col from B;

COL

(A – B) + (B – A)

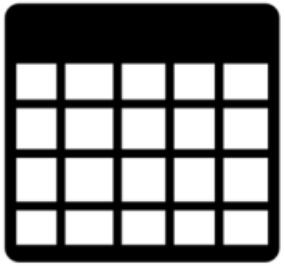
A



?

=

B



```
select col from A
minus
select col from B;
```

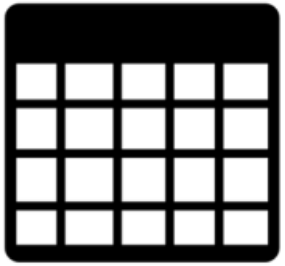
COL

```
select col from B
minus
select col from A;
```

COL

$(A - B) + (B - A)$

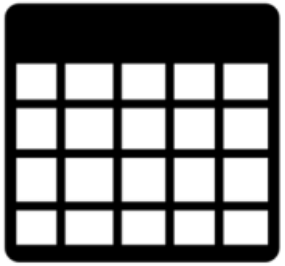
A

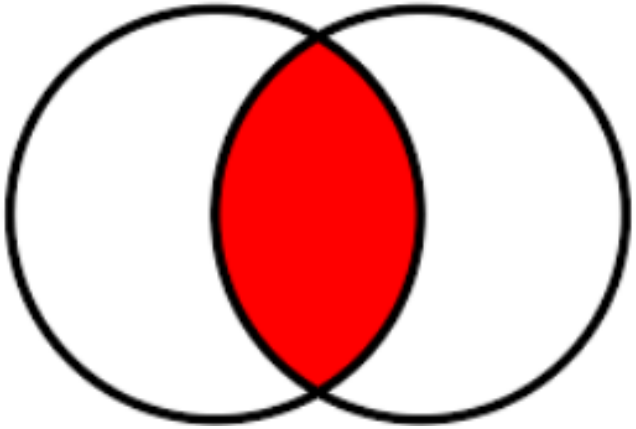


?

=

B





Hash Code Primer

select STANDARD_HASH ('foo bar','MD5') MD5 from dual;

MD5
327B6F07435811239BC47E1544353273

select STANDARD_HASH ('foo bar','SHA1') SHA1 from dual;

SHA1
3773DEA65156909838FA6C22825CAFE090FF8030

select STANDARD_HASH ('foo bar','SHA256') SHA256 from dual;

SHA256
FBC1A9F858EA9E177916964BD88C3D37B91A1E84412765E29950777F265C4B75

select STANDARD_HASH ('foo bar','SHA512') SHA512 from dual;

SHA512	....	
65019286222ACE418F742556366F9B9DA5AAF6797527D2F0CBA5BFE6B2F8ED		4F378B03F10090CE7

Motivation

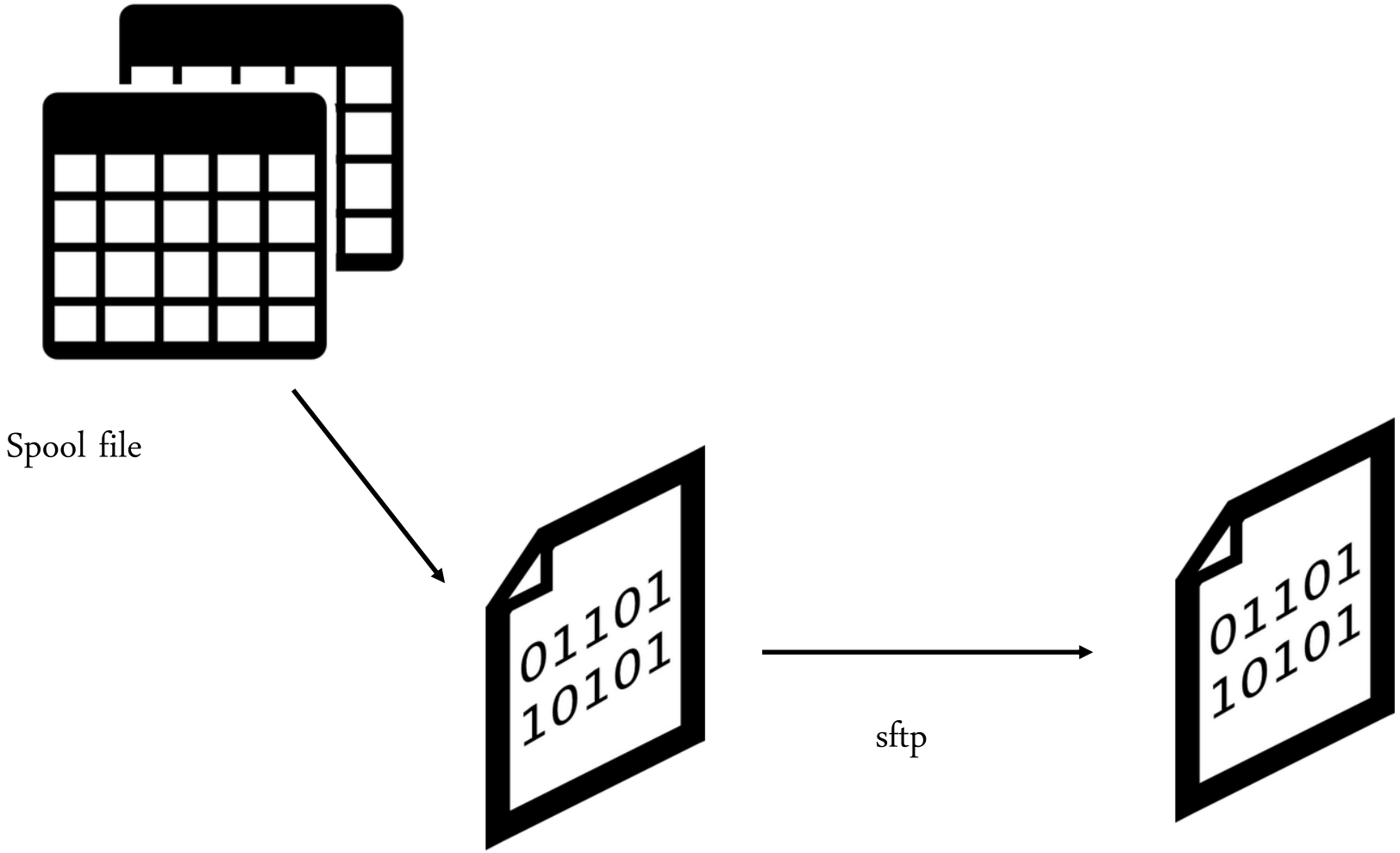


Image source http://clipart-library.com/clipart/

Motivation

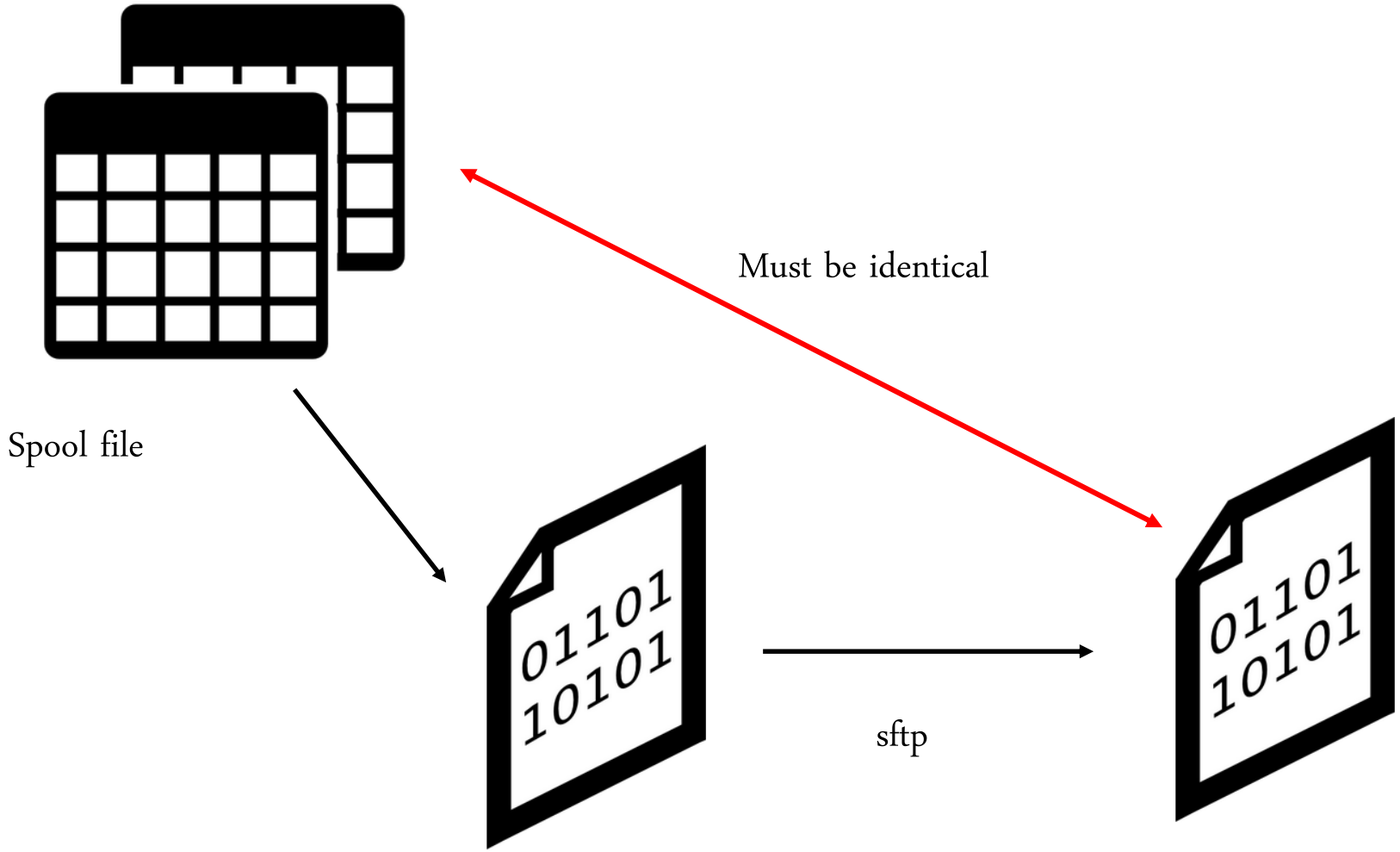


Image source http://clipart-library.com/clipart/

Is Hash Code Unique?

Image source http://clipart-library.com/clipart/

Hash Code of a File

```
# cat foo.txt
foo
bar

# md5sum foo.txt
f47c75614087a8dd938ba4acff252494  foo.txt
```

Motivation

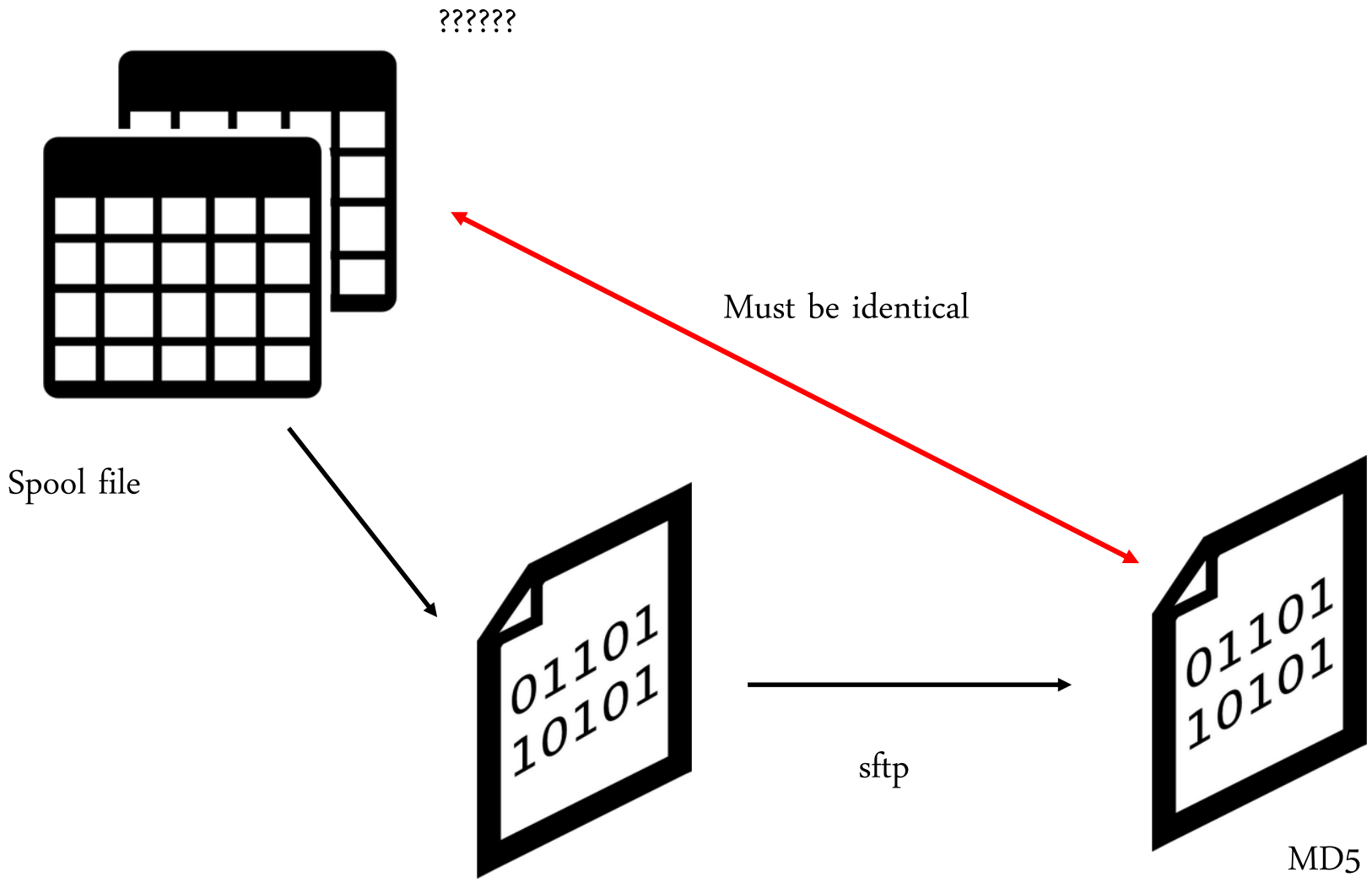


Image source http://clipart-library.com/clipart/

Hash Code of Table

```
select standard_hash('foo'||chr(10)||'bar'||chr(10), 'MD5') MD5
from dual;
```

MD5
F47C75614087A8DD938BA4ACFF252494

Hash Code of Table

```
select standard_hash('foo'||chr(10)||'bar'||chr(10), 'MD5') MD5
from dual;
```

MD5
F47C75614087A8DD938BA4ACFF252494

```
# md5sum foo.txt
f47c75614087a8dd938ba4acff252494   foo.txt
```

MD5 Recalculated

```
MessageDigest digest =
    MessageDigest.getInstance("MD5")
byte[] md5hash

conn.eachRow
   ('select txt from MY_TABLE order by id')
     {
       digest.update
       (it.txt.getBytes(StandardCharsets.UTF_8))
     }

md5hash = digest.digest();
println md5hash.encodeHex().toString()
```

MD5 Recalculated

```
MessageDigest digest =
    MessageDigest.getInstance("MD5")
by

conn.eachRow
    ('select txt from MY_TABLE order by id')
      {
        digest.update
        (it.txt.getBytes(StandardCharsets.UTF_8))
      }

md5hash = digest.digest();
println md5hash.encodeHex().toString()
```

MD5 Recalculated

```
MessageDigest digest =
    MessageDigest.getInstance("MD5")
byte[] md5hash

conn.eachRow
  select txt from MY_TABLE order by id
    {
      digest.update
                                        UTF_8))
    }       (txt.getBytes(UTF_8))

md5hash = digest.digest();
println md5hash.encodeHex().toString()
```

MD5 Recalculated


```
MessageDigest digest =
    MessageDigest.getInstance("MD5")
byte[] md5hash

conn.eachRow
   ('select txt from MY_TABLE order by id')
     {
       digest.update
       (it.txt.getBytes(StandardCharsets.UTF_8))
     }

md5hash = digest.digest();
println md5hash.encodeHex().toString()
```

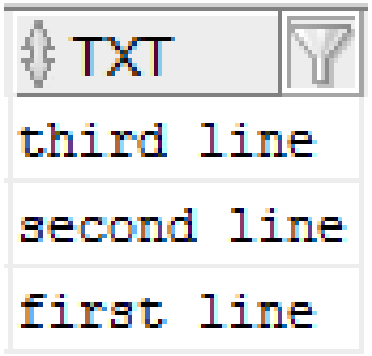
Two Problems

- Not a PL/SQL implementation
- Sorting required

Order of Data

For a file the order is critical

```
first line
second line
third line
```

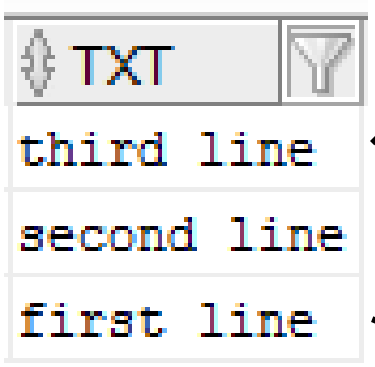


XOR

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

A XOR B = B XOR A

XOR Hash Combination



4B6A1E62E6B79642AB1310DB97128F56

XOR

63D4C06D0DD227A33B7CCB9CAA856828

XOR

098BA91C08776B95743BF9CE9C09E65E

21357713E312DA74E4542289A19E0120

User Defined Aggregate Function MD5_XOR

```
MEMBER FUNCTION ODCIAggregateIterate(self IN OUT md5_xor_type,
                                     VALUE IN VARCHAR2 )
  RETURN NUMBER
  IS
 my_hash RAW(16);
 BEGIN
  BEGIN
     select  standard_hash(VALUE, 'MD5') into my_hash from dual;
  END; -- add exception handling
  IF self.md5Hash IS NULL THEN
     self.md5Hash := my_hash;
  ELSE
     self.md5Hash := UTL_RAW.BIT_XOR(self.md5Hash,my_hash);
  END IF;
  RETURN ODCIConst.Success;
 END;
```

User Defined Aggregate Function MD5_XOR

```
MEMBER FUNC ODCIAggregateIterate N OUT md5_xor_type,
                                   VALUE IN VARCHAR2 )
  RETURN NUMBER
  IS
 my_hash RAW(16);
 BEGIN
  BEGIN
     select standard_hash(VALUE, 'MD5') from dual;
  END; -- add exception handling
  IF self.md5Hash IS NULL THEN
     self.md5Hash := my_hash;
  ELSE
     UTL_RAW.BIT_XOR(self.md5Hash,my_hash);
  END IF;
  RETURN ODCIConst.Success;
 END;
```

MD5_XOR

```
select MD5_XOR(col1) hash_md5 from cmp1;
```

HASH_MD5
80FE13F9B1C93DF72DA868F64128CDFF

MD5_XOR Column Concatenation

```
select MD5_XOR(col1 || col2) hash_md5 from cmp1;
```

HASH_MD5
33C4A7BDCF007D9A94B0D2294F1977CE

MD5_XOR Column Concatenation

Problem with NULLs

NULL || 'A' .... 'A' || NULL

Use delimiter (not contained in the data)

select MD5_XOR(col1 ||'.'|| col2) hash_md5 from cmp1;

HASH_MD5
DBBB4FCEA29B718DDEE24E033FF81573

A = B ?

```
select   MD5_XOR(col) md5
from A;
```

```
select   MD5_XOR(col) md5
from B;
```

A = B ?

```
select   MD5_XOR(col) md5
from A;
```

MD5
ACBD18DB4CC2F85CEDEF654FCCC4A4D8

```
select   MD5_XOR(col) md5
from B;
```

MD5
37B51D194A7513E45B56F6524F2D51F2

A = B ?

```
select   MD5_XOR(col) md5
from A;
```

MD5
ACBD18DB4CC2F85CEDEF654FCCC4A4D8

```
select * from A;
```

COL
foo
bar
bar

```
select   MD5_XOR(col) md5
from B;
```

MD5
37B51D194A7513E45B56F6524F2D51F2

```
select * from B;
```

A = B ?

```
select   MD5_XOR(col) md5
from A;
```

MD5
ACBD18DB4CC2F85CEDEF654FCCC4A4D8

```
select * from A;
```

COL
foo
bar
bar

```
select   MD5_XOR(col) md5
from B;
```

MD5
37B51D194A7513E45B56F6524F2D51F2

```
select * from B;
```

COL
foo
foo
bar

Possible Use Cases

Observe Immutable Tables

Remote Compare

Observe Immutable Tables

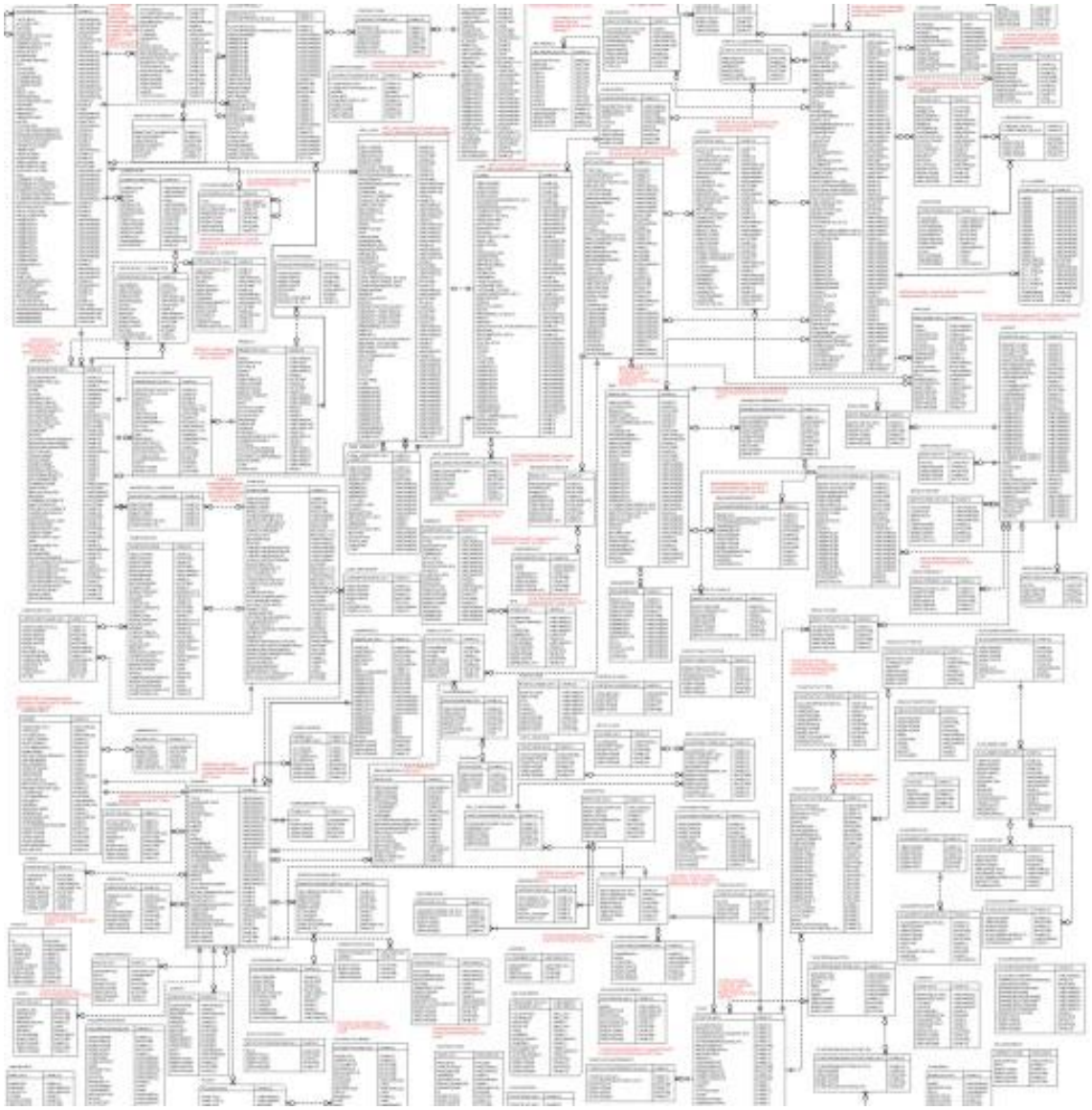


Image Source: https://www.safaribooksonline.com/library/view/programming-the-semantic/9780596802141/ch01.html

Observe Immutable Tables

TABLE_NAME	TIME_STAMP	MD5_HASH
Table1	31.05.18 13:40:07	ACBD18DB4CC2F85CEDEF654FCCC4A4D8
Table2	31.05.18 13:40:07	37B51D194A7513E45B56F6524F2D51F2
Table3	31.05.18 13:40:07	73FEFFA4B7F6BB68E44CF984C85F6E88

Observe Immutable Tables

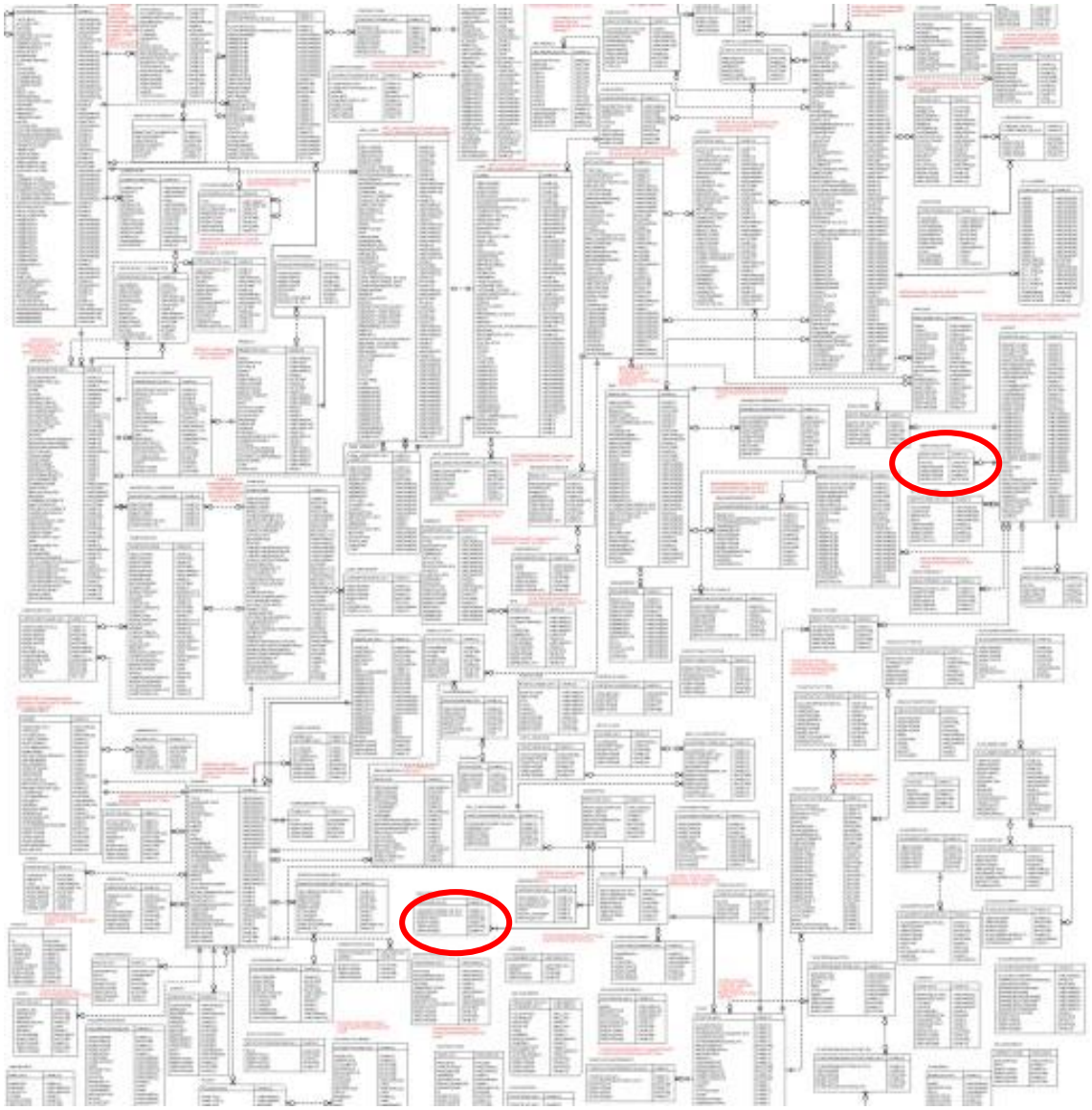
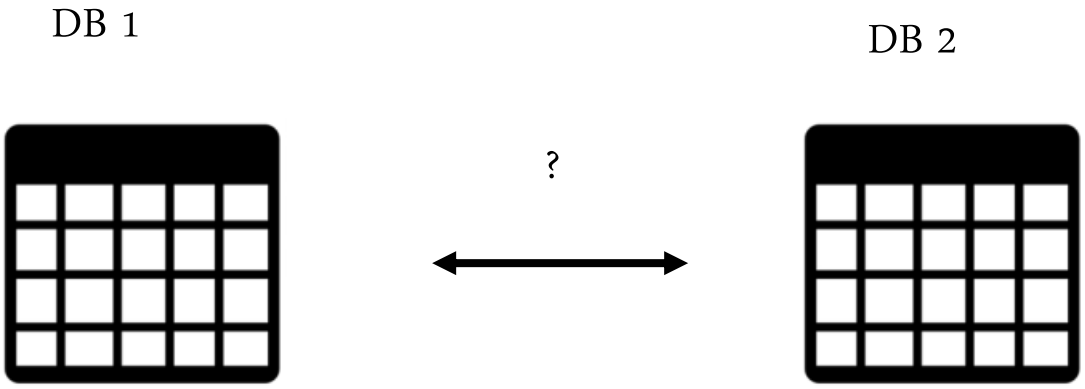


Image Source: https://www.safaribooksonline.com/library/view/programming-the-semantic/9780596802141/ch01.html

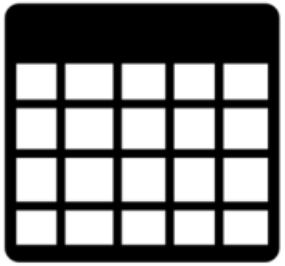
Remote Compare



Compare two large tables in different databases without passing much data over network

Remote Compare

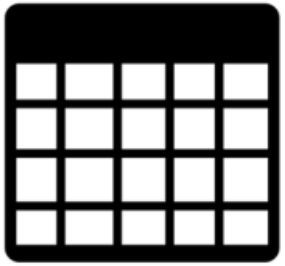
DB 1



MD5_HASH
ACBD18DB4CC2F85CEDEF654FCCC4A4D8

?

DB 2



MD5_HASH
4B6A1E62E6B79642AB1310DB97128F56

Remote Compare

DB 1

```
select grp_id, MD5_XOR(col1||col2) hash_md5
from cmp1
group by grp_id
```

GRP_ID	HASH_MD5
0	49E24489B738E2EDD8B5870C9DD6F69A
1	0D08057BAD83FECDF1139C44F7FD3049
2	97C4D33DC611FD68FEE1157FA14AA87F
3	7658E1C495C26B43047A6D6DFD3B5487
4	8D619C8BE12D7E9CA0450E7781F5C6E3
5	92003DF5DE99F7CA8F6E247C3FEF9EF6
6	1DAF3E58FE1FC1BC0AA8613BF549A424
7	45ACA5382828BE50436B3E254A0E5DD4
8	C57651D4B5D05BA8C63F20BD24D5B6BD
9	14A6BF7CDA3B5A83E75AE4DB5CCB5ABD

Remote Compare

DB 2

```
select grp_id, MD5_XOR(col1||col2) hash_md5
from cmp2@REMOTE
group by grp_id
```

GRP_ID	HASH_MD5
0	49E24489B738E2EDD8B5870C9DD6F69A
1	BBDF62A9556EA3EC5558921C71953CF0
2	97C4D33DC611FD68FEE1157FA14AA87F
3	7658E1C495C26B43047A6D6DFD3B5487
4	8D619C8BE12D7E9CA0450E7781F5C6E3
5	92003DF5DE99F7CA8F6E247C3FEF9EF6
6	1DAF3E58FE1FC1BC0AA8613BF549A424
7	45ACA5382828BE50436B3E254A0E5DD4
8	C57651D4B5D05BA8C63F20BD24D5B6BD
9	14A6BF7CDA3B5A83E75AE4DB5CCB5ABD

Only hash codes are passed over the network

Remote Compare - Final View

GRP_ID	IS_IDENTICAL	C1_HASH_MD5	C2_HASH_MD5
0	Y	49E24489B738E2EDD8B5870C9DD6F69A	49E24489B738E2EDD8B5870C9DD6F69A
1	N	0D08057BAD83FECDF1139C44F7FD3049	BBDF62A9556EA3EC5558921C71953CF0
2	Y	97C4D33DC611FD68FEE1157FA14AA87F	97C4D33DC611FD68FEE1157FA14AA87F
3	Y	7658E1C495C26B43047A6D6DFD3B5487	7658E1C495C26B43047A6D6DFD3B5487
4	Y	8D619C8BE12D7E9CA0450E7781F5C6E3	8D619C8BE12D7E9CA0450E7781F5C6E3
5	Y	92003DF5DE99F7CA8F6E247C3FEF9EF6	92003DF5DE99F7CA8F6E247C3FEF9EF6
6	Y	1DAF3E58FE1FC1BC0AA8613BF549A424	1DAF3E58FE1FC1BC0AA8613BF549A424
7	Y	45ACA5382828BE50436B3E254A0E5DD4	45ACA5382828BE50436B3E254A0E5DD4
8	Y	C57651D4B5D05BA8C63F20BD24D5B6BD	C57651D4B5D05BA8C63F20BD24D5B6BD
9	Y	14A6BF7CDA3B5A83E75AE4DB5CCB5ABD	14A6BF7CDA3B5A83E75AE4DB5CCB5ABD

Remote Compare - Final View

GRP_ID	IS_IDENTICAL	C1_HASH_MD5	C2_HASH_MD5
0	Y	49E24489B738E2EDD8B5870C9DD6F69A	49E24489B738E2EDD8B5870C9DD6F69A
1	N	0D08057BAD83FECDF1139C44F7FD3049	BBDF62A9556EA3EC5558921C71953CF0
2	Y	97C4D33DC611FD68FEE1157FA14AA87F	97C4D33DC611FD68FEE1157FA14AA87F
3	Y	7658E1C495C26B43047A6D6DFD3B5487	7658E1C495C26B43047A6D6DFD3B5487
4	Y	8D619C8BE12D7E9CA0450E7781F5C6E3	8D619C8BE12D7E9CA0450E7781F5C6E3
5	Y	92003DF5DE99F7CA8F6E247C3FEF9EF6	92003DF5DE99F7CA8F6E247C3FEF9EF6
6	Y	1DAF3E58FE1FC1BC0AA8613BF549A424	1DAF3E58FE1FC1BC0AA8613BF549A424
7	Y	45ACA5382828BE50436B3E254A0E5DD4	45ACA5382828BE50436B3E254A0E5DD4
8	Y	C57651D4B5D05BA8C63F20BD24D5B6BD	C57651D4B5D05BA8C63F20BD24D5B6BD
9	Y	14A6BF7CDA3B5A83E75AE4DB5CCB5ABD	14A6BF7CDA3B5A83E75AE4DB5CCB5ABD

Merkle tree

Performance Considerations

Two news – which one first?

Performance Considerations

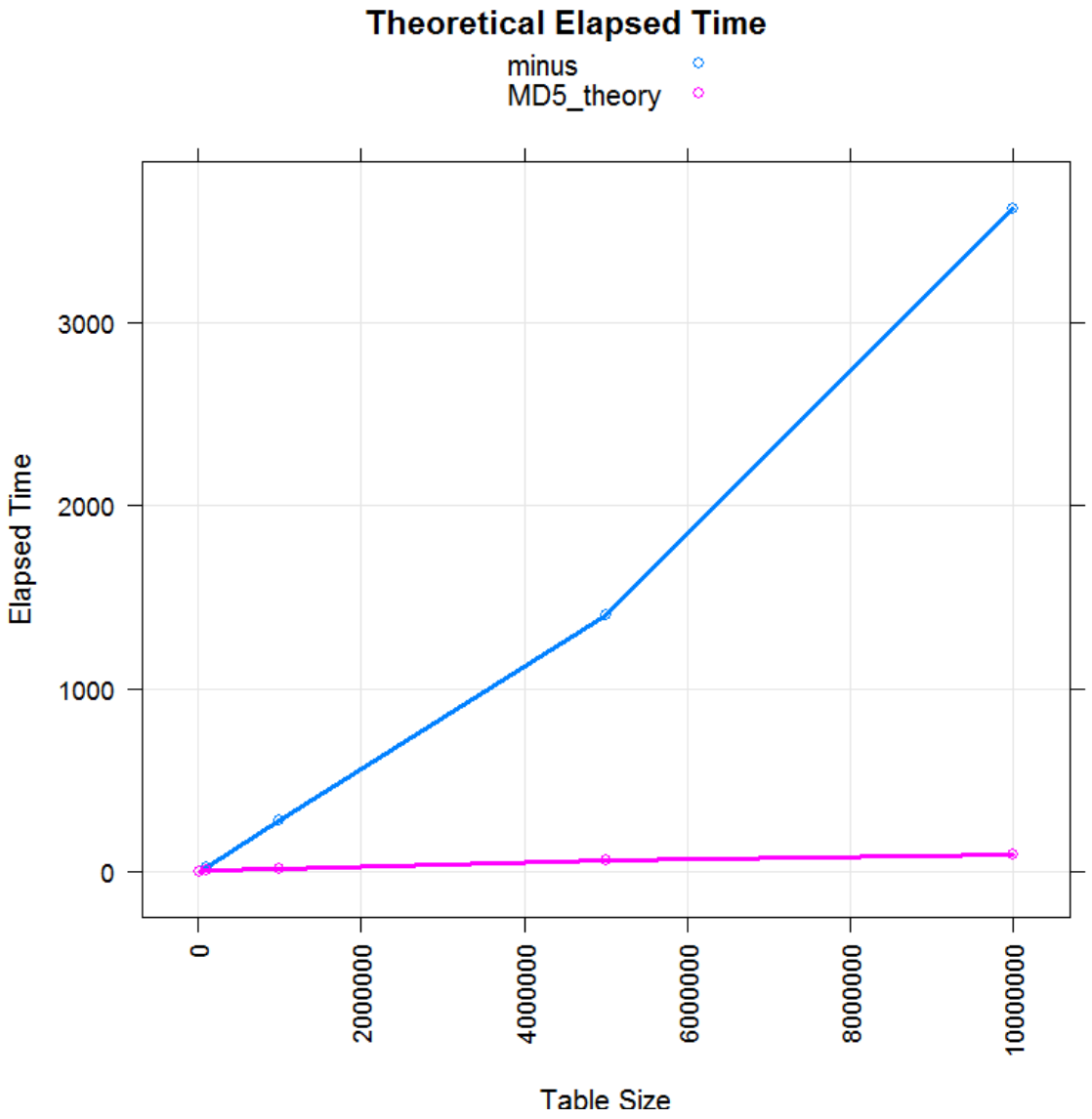
Hash Code is calulated in linear time

O(N)

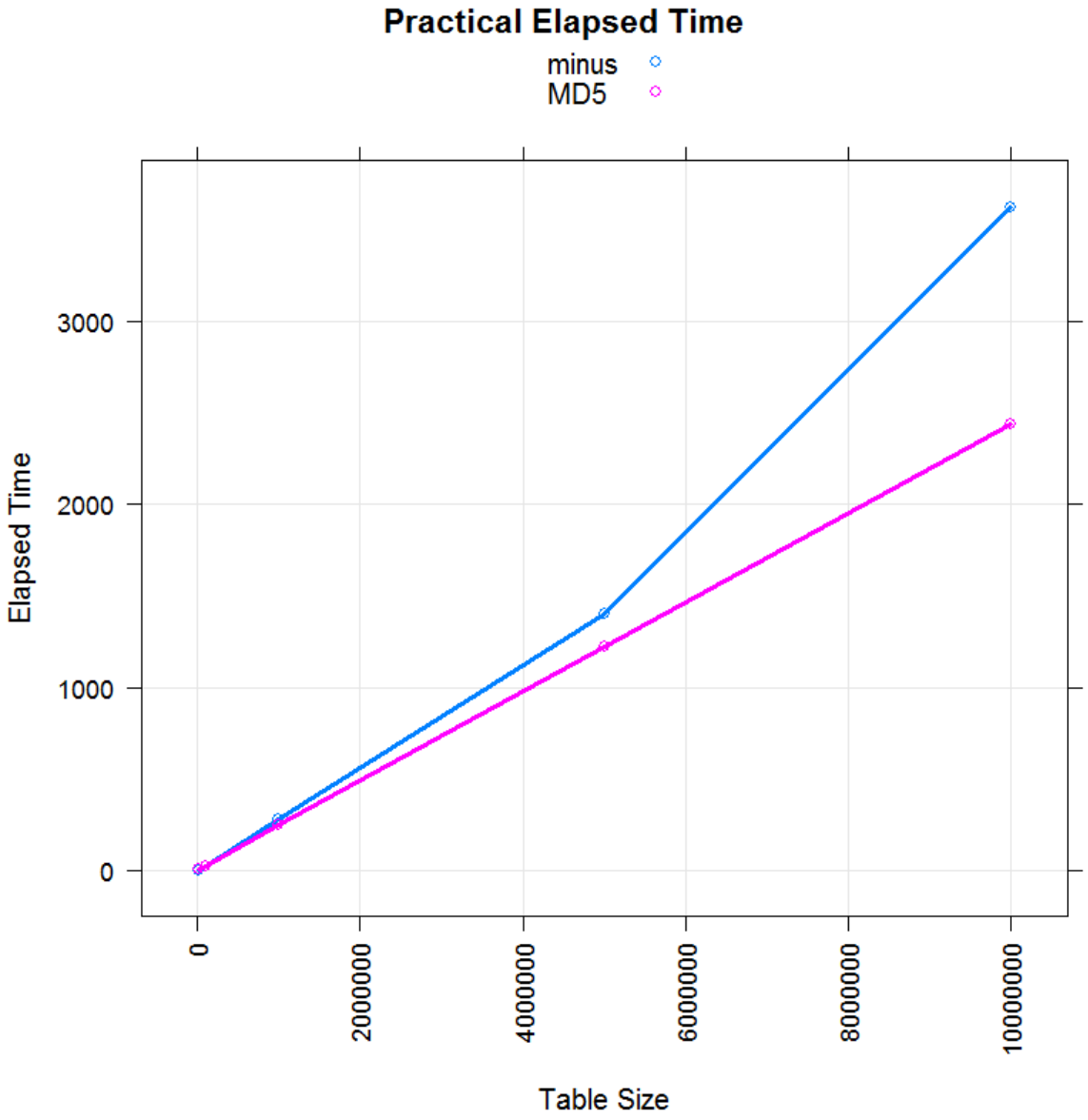
Set operations (MINUS, UNION) need sort

O(N * log(N))

Performance Considerations



Performance Considerations



Oracle Native Implementation

Aggregate function supparty all hash types

XOR_hash(col1, 'MD5')

Including column concatenation

XOR_hash(col1, col2, col3, 'MD5')

Oracle Package DBMS_SQLHASH

```
select DBMS_SQLHASH.GETHASH(
            'select col from A order by col',
            2 -- 'HASH_MD5'
      ) MD5
from dual;
```

MD5
A62A79A54326A16DED331E8B1E847C00

Oracle Database Ideas

More ideas in Database Ideas

Aggregate Function to Calculate a Hash Code for a Whole Table

Created on 17-Dec-2017 20:06 by nemecj - Last Modified: 25-Mar-2018 10:14

0 You have not voted. ACTIVE

Summary

MINUS, UNION

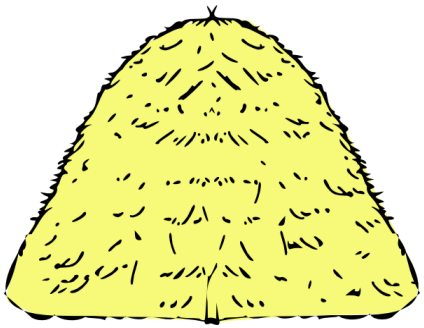
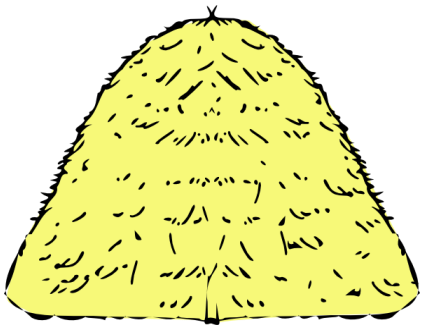
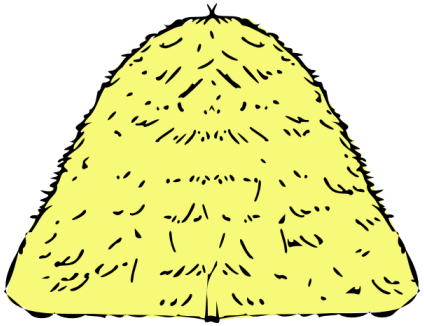
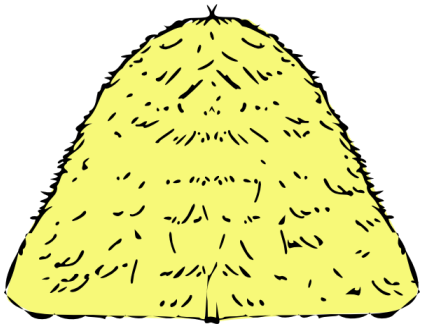
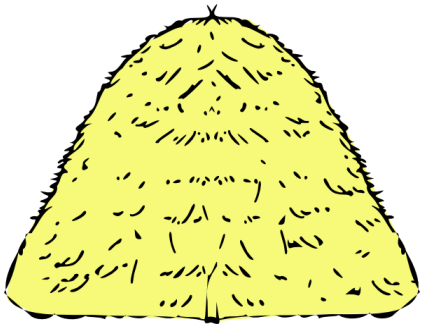
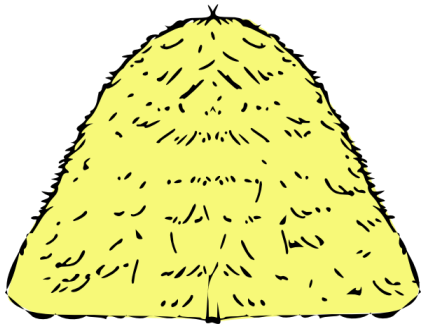
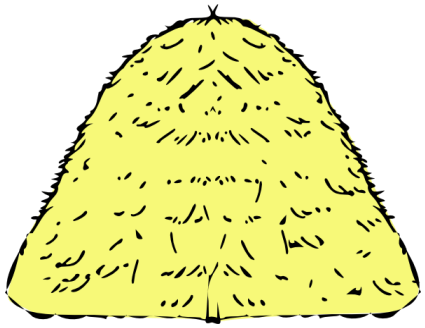
HASH

COUNT, SUM

Image source http://clipart-library.com

Image source Claude Monet

Summary

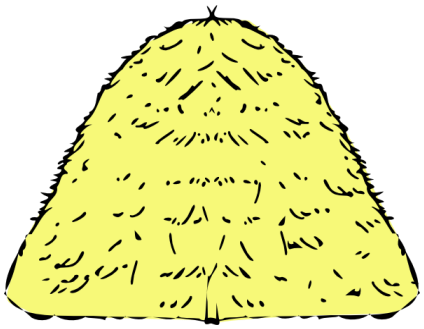


Image source http://clipart-library.com

Image source Claude Monet

Summary

Image source http://clipart-library.com

Summary

References

White Paper

Hash Code Based Identity in the Database

http://www.db-nemec.com/MD5/CompareTablesUsingMD5Hash.html

Oracle Ideas

Aggregate Function to Calculate a Hash Code for a Whole Table

https://community.oracle.com/ideas/20275

Q & A

?

Verifying Dovnload File

2.6.1 Verifying the MD5 Checksum

```
shell> md5sum mysql-standard-5.0.96-linux-i686.tar.gz
aaab65abbec64d5e907dcd41b8699945  mysql-standard-5.0.96-linux-i686.tar.gz
```

If my copy of file has the same hash as the official version
the files are identical

https://docs.oracle.com/cd/E17952_01/mysql-5.0-en/verifying-md5-checksum.html